AD-A062 563 WEAPONS SYSTEMS RESEARCH LAB ADELAIDE (AUSTRALIA) F/G 19/4

THE SOFTWARE FOR THE CAPTIVE TRAJECTORY YAWMETER SYSTEM.(U)

APR 78 I C HERON, G R BISHOP

UNCLASSIFIED WSRL-0005-TR NL

1 OF 2

AD A062563

(12) LEVEL II

WSRL-0005-TR

AR-001-176

AD A062563

DEPARTMENT OF DEFENCE

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

WEAPONS SYSTEMS RESEARCH LABORATORY

DEFENCE RESEARCH CENTRE SALISBURY
SOUTH AUSTRALIA

TECHNICAL REPORT

WSRL-0005-TR

THE SOFTWARE FOR THE CAPTIVE TRAJECTORY YAWMETER SYSTEM (U)

I.C. HERON and G.R. BISHOP

COPY No. 83

APRIL 1978

UNCLASSIFIED

AR-001-176

DEPARTMENT OF DEFENCE

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

WEAPONS SYSTEMS RESEARCH LABORATORY

9 TECHNICAL REPORT

WSRL-0005-TR

16 THE SOFTWARE FOR THE CAPTIVE TRAJECTORY YAWMETER SYSTEM (U)

10 I.C. Heron ~~and~~ G.R. Bishop

11 Apr 78 12 99p.

S U M M A R Y

This report describes the mathematical analysis and software associated with the Captive Trajectory Yawmeter System (CTYS), which is used in Aeroballistics Division, WSRL to simulate store separation from aircraft. The CTYS uses a yawmeter probe attached to a traverse rig to measure flow conditions from which store loads are derived, and thus differs from the traditional Captive Trajectory System in which store loads are measured directly. The CTYS is therefore applicable to small continuous flow wind tunnels. Computer listings of all programs are included. (U)

14 WSRL-0005-TR

Approved for Public Release

POSTAL ADDRESS: Chief Superintendent, Weapons Systems Research Laboratory, Box 2151, G.P.O., Adelaide, South Australia, 5001.

410 919

UNCLASSIFIED

DOCUMENT CONTROL DATA SHEET

Security classification of this page: UNCLASSIFIED

1 DOCUMENT NUMBERS	
AR Number:	AR-001-176
Report Number:	WSRL-0005-TR
Other Numbers:	

2 SECURITY CLASSIFICATION	
a. Complete Document:	Unclassified
b. Title in Isolation:	Unclassified
c. Summary in Isolation:	Unclassified

3 TITLE

THE SOFTWARE FOR THE CAPTIVE TRAJECTORY YAWMETER SYSTEM (U)

4 PERSONAL AUTHOR(S):

I.C. Heron and G.R. Bishop

5 DOCUMENT DATE:

April 1978

6 6.1 TOTAL NUMBER OF PAGES

6.2 NUMBER OF REFERENCES: 11

7 7.1 CORPORATE AUTHOR(S):

Weapons Systems Research Laboratory

7.2 DOCUMENT SERIES AND NUMBER

Weapons Systems Research Laboratory 0005-TR

8 REFERENCE NUMBERS	
a. Task:	DST 77/046
b. Sponsoring Agency:	DS70

9 COST CODE:

539AC345

10 IMPRINT (Publishing organisation)

Defence Research Centre Salisbury

11 COMPUTER PROGRAM(S) (Title(s) and language(s))

12 RELEASE LIMITATIONS (of the document):

Approved for Public Release

12.0	OVERSEAS	NO		P.R.	1	A		B		C		D		E	

Security classification of this page: UNCLASSIFIED

Security classification of this page: UNCLASSIFIED

13 ANNOUNCEMENT LIMITATIONS (of the information on these pages):

No limitation

14 DESCRIPTORS:

a. EJC Thesaurus Terms

External stores
Subsonic flow
Transonic flow
Wind tunnels
Computer systems programs
Computerized simulation
Equations of motion
Trajectories
Euler equations of motion
Aerodynamics

b. Non-Thesaurus Terms

Store separation simulation

15 COSATI CODES:

0103
2004
1201

16 LIBRARY LOCATION CODES (for libraries listed in the distribution):

17 SUMMARY OR ABSTRACT:
(if this is security classified, the announcement of this report will be similarly classified)

This report describes the mathematical analysis and software associated with the Captive Trajectory Yawmeter System (CTYS), which is used in Aeroballistics Division, WSRL to simulate store separation from aircraft. The CTYS uses a yawmeter probe attached to a traverse rig to measure flow conditions from which store loads are derived, and thus differs from the traditional Captive Trajectory System in which store loads are measured directly. The CTYS is therefore applicable to small continuous flow wind tunnels. Computer listings of all programs are included. (U)

Security classification of this page: UNCLASSIFIED

TABLE OF CONTENTS

LIST OF APPENDICES

LIST OF FIGURES

1. INTRODUCTION

The Captive Trajectory Yawmeter System (CTYS) is the method used in Aeroballistics Division, WSRL for simulating store separation from aircraft.

The initial plan was reported in reference 1, preliminary validations are described in reference 2, and progress reports were given in references 3 and 4. This report describes the mathematical background and the software which drives the system.

Section 2 provides the details of all coordinate systems used, defines Euler angles and derives the equations of motion for the six-degree-of-freedom program. Section 3 outlines the methods for determining the forces and moments on the store, while Section 4 gives details of the system and software.

The CTYS resides on a UNICHANNEL-15 (PDP-15/PDP-11) dual processor computer which is attached to Aeroballistics Division's S1 wind tunnel, a continuous flow tunnel which can operate in a speed range from about 100 m/s to Mach 1 and from Mach 1.4 to Mach 2.8. A full description of the system can be found in reference 5 and its associated references.

In view of the small size (0.4 m x 0.4 m) of the working section, the traditional method of using a model of the store mounted on a sting has been replaced by the technique of measuring the flow at twelve stations along the store centre-line for each trajectory point computed. Thus, the store loads are derived from measurements of the flow field and are not directly measured. Aircraft models are typically 1/50th scale.

The flow measurements are made by a yawmeter probe mounted on a computer-controlled four-degree-of-freedom traverse rig, the roll axis being inoperative for this application. The probe measures pitot pressure, manifold pressure and two differential pressures, which are sufficient to determine the flow properties at each measurement station.

The CTYS then determines forces and moments by:

(a) selecting the most appropriate result from data which have been previously acquired on an approximately 1/8th scale model of the store in uniform flow, and

(b) calculating increments to the store-loading due to the known non-uniform flow, as measured on-line by the probe.

Section 3 gives a full account of this method.

In other respects the CTYS is analogous to the Captive Trajectory System, being simply a point prediction technique using a fourth-order Runge Kutta integration in time. One significant feature of the method of load estimation is that, as the store moves away from the aircraft, the flow non-uniformities decrease and the corrections to the measured coefficients tend to zero. At this stage the aircraft does not influence the trajectory any more and the trajectory can then be continued to impact, or as far as desired, without further tunnel information.

The time required to complete an average trajectory is about 30 min. Although the system has been streamlined as much as possible, there are many inherent delays such as traverse rig travel time and yawmeter probe settling delay, the latter being approximately 5 s. Many accesses to the various disk data stores must be made for each trajectory point, but these are performed using Direct Access Input/Output, mainly while the rig is traversing to the next point, and therefore do not generally delay the process.

Results to date are promising and it is expected that full scale trajectories can be simulated quite successfully.

2. MATHEMATICAL ANALYSIS

2.1 Definition of coordinate systems and Euler angles

The Captive Trajectory Yawmeter System uses two distinct sets of coordinate axes. The first set refers to the real world and consists of earth axes, aircraft axes and bomb axes as shown in figure 1.

The aircraft is assumed to move in a circle in the $X_E Z_E$ plane at a constant tangential velocity V_∞ and with centripetal acceleration μg. Its initial attitude is specified by a dive angle (DIVE) and an angle of attack (ATTACK). Also, since the Aircraft Z axis (Z_A) points in a downward direction and the Earth Z axis (Z_E) points upwards, the roll of the aircraft with respect to earth axes is π radians and its pitch is given by

$$\text{APITCH} = -(\text{DIVE}-\text{ATTACK}).$$

The aircraft maintains this trajectory throughout, where only the dive angle changes with time. The value of μ in the centripetal accelerations may also be negative.

The CTYS also defines a set of wind tunnel coordinates as shown in figure 2. The flow (V_∞) is parallel to the tunnel walls whereas the aircraft and bomb axes are inverted, the aircraft being fixed to the tunnel wall at an angle of attack; the bomb position is defined by a yawmeter probe attached to a four-degree-of-freedom traverse rig. The wind tunnel Z axis (Z_T) points downwards thus providing a certain parity with the real world system of figure 1.

Throughout the analysis, the orientation of one axis system with respect to another is described by Euler angles for roll, pitch and yaw. For example, to convert from aircraft axes to bomb axes:

(i) Roll through an angle φ about the aircraft X axis (X_A) to give a new system (X',Y',Z'), where X' = X_A. This coordinate transformation is described by the 3 x 3 matrix

$$C_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{pmatrix} \qquad (1)$$

(ii) Then pitch through an angle θ about the Y' axis to give the system (X",Y",Z"), where Y" = Y'. The corresponding transformation matrix is

$$C_2 = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \qquad (2)$$

(iii) Finally, yaw through an angle β about the Z" axis to give bomb axes (X_B,Y_B,Z_B), where Z_B = Z". This corresponds to the transformation

$$C_3 = \begin{pmatrix} \cos\beta & \sin\beta & 0 \\ -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The total transformation is thus represented by the direction cosine matrix $C = C_3C_2C_1$, which is given by

$$C = \begin{pmatrix} \cos\theta\cos\beta & \sin\varphi\sin\theta\cos\beta + \cos\varphi\sin\beta & -\cos\varphi\sin\theta\cos\beta + \sin\varphi\sin\beta \\ -\cos\theta\sin\beta & -\sin\varphi\sin\theta\sin\beta + \cos\varphi\cos\beta & \cos\varphi\sin\theta\sin\beta + \sin\varphi\cos\beta \\ \sin\theta & -\sin\varphi\cos\theta & \cos\varphi\cos\theta \end{pmatrix} \tag{4}$$

Thus, if $\underline{a}$ and $\underline{b}$ are vectors in aircraft axes and bomb axes respectively, then

$$\underline{b} = C\underline{a} \tag{5}$$

2.2 The relation between Euler rates and bomb angular rates

Let p,q and r be the angular rates of rotation of the rotating axis system about its own X,Y and Z axes respectively. To calculate the relationship between these angular rates and the Euler rates $\dot{\varphi}$, $\dot{\theta}$ and $\dot{\beta}$, it is necessary to use the results of the previous section.

Firstly, a roll of φ about the aircraft X axis gives rise to a rate vector $(\dot{\varphi},0,0)$ in aircraft axes. This produces a component

$$\underline{\omega}_1 = C_3\ C_2\ C_1 \begin{pmatrix} \dot{\varphi} \\ 0 \\ 0 \end{pmatrix} \tag{6}$$

in the rotating axis system, where the matrices C_1, C_2 and C_3 are defined by equations (1), (2) and (3) respectively.

Secondly, a pitch of θ about the Y' axis (see Section 2.1) produces a rate component

$$\underline{\omega}_2 = C_3\ C_2 \begin{pmatrix} 0 \\ \dot{\theta} \\ 0 \end{pmatrix} \tag{7}$$

in the rotating axis system.

Finally a yaw of β about the Z" axis produces a rate component

$$\underline{\omega}_3 = C_3 \begin{pmatrix} 0 \\ 0 \\ \dot{\beta} \end{pmatrix} \tag{8}$$

in the rotating axis system.

Summing the contributions of equations (6), (7) and (8), we obtain

$$p = \dot{\theta} \sin\beta + \dot{\varphi} \cos\theta \cos\beta \tag{9(a)}$$

$$q = \dot{\theta} \cos\beta - \dot{\varphi} \cos\theta \sin\beta \tag{9(b)}$$

and

$$r = \dot{\beta} + \dot{\varphi} \sin\theta . \tag{9(c)}$$

Conversely,

$$\dot{\varphi} = (p \cos\beta - q \sin\beta)/\cos\theta \tag{10(a)}$$

$$\dot{\theta} = p \sin\beta + q \cos\beta \tag{10(b)}$$

and

$$\dot{\beta} = r - \tan\theta \, (p \cos\beta - q \sin\beta). \tag{10(c)}$$

Note that equations (10) have a singularity at $\theta = \frac{\pi}{2}$. This fact will be of significance later.

2.3 Rate of change of the Euler angle transformation matrix

It will be useful to calculate the rate of change of the direction cosine matrix C depicted by equation (4).

Let F be a fixed axis system and B be an axis system which is rotating at a rate ω with respect to F.

Let $\underline{b}$ be a vector in B whose coordinates in the fixed system F are given by $\underline{f}$. Then

$$\underline{b} = C\underline{f} \tag{11}$$

where C is a 3 x 3 orthogonal direction-cosine matrix.
Now, it is well known that

$$\dot{\underline{f}} = \dot{\underline{b}} + \underline{\omega} \times b, \tag{12}$$

the second term deriving from the rotation of axis system B. Now since $\underline{\omega}$ is a vector in F, then we can define a vector $\underline{\omega}' = (p,q,r)$, where p, q and r are the angular rates of rotation of B about its own X, Y and Z axes respectively. Further, since $\underline{\omega}'$ is a vector in B, then

$$\underline{\omega}' = C\underline{\omega} . \tag{13}$$

Equations (12) and (13) then give

$$\dot{\underline{f}} = \dot{\underline{b}} + C^{-1} \underline{\omega}' \times \underline{b}. \tag{14}$$

Now $\underline{\omega}' \times \underline{b}$ can be expressed as the matrix product $-\Omega\,\underline{b}$, where

$$\Omega = \begin{pmatrix} 0 & r & -q \\ -r & 0 & p \\ q & -p & 0 \end{pmatrix} \tag{15}$$

Hence, equations (14) and (15) yield the relations

$$\dot{\underline{f}} = \dot{\underline{b}} - C^{-1}\,\Omega\,\underline{b}. \tag{16}$$

To calculate $\dot{C}$, the rate of change of the direction cosine matrix C, let $\underline{b}$ be a vector which is fixed in the rotating system B, so that

$$\dot{\underline{b}} = \underline{0} \tag{17}$$

and

$$\dot{\underline{f}} = -C^{-1}\,\Omega\,\underline{b}. \tag{18}$$

Differentiation of equation (11) with respect to time gives

$$\dot{\underline{b}} = \dot{C}\underline{f} + C\dot{\underline{f}}. \tag{19}$$

Then substitution of equations (11), (17) and (18) into equation (19) yields

$$\begin{aligned} \underline{0} &= \dot{C}\underline{f} + C(-C^{-1}\,\Omega\,\underline{b}) \\ &= \dot{C}\underline{f} - \Omega\,C\,\underline{f}. \end{aligned}$$

That is,

$$(\dot{C} - \Omega\,C)\underline{f} = \underline{0} \tag{20}$$

Since $\underline{f}$ is an arbitrary vector, we can deduce that

$$\dot{C} = \Omega\,C. \tag{21}$$

2.4 Trajectory of aircraft

As noted in Section 2.1, the aircraft describes a circular trajectory which is either convex to the origin (μ positive) or concave to the origin (μ negative). The centripetal acceleration (μg) is assumed to include the gravitational component.

Thus, at all times

$$\frac{V_\infty^2}{\rho} = \mu g \tag{22}$$

and

$$V_\infty = -\rho \frac{d}{dt} (\text{DIVE}), \tag{23}$$

where ρ is the radius of curvature and DIVE is the dive angle, a function of time.
Eliminating ρ from equations (22) and (23) gives

$$\text{DIVE}(t) = \text{DIVE}_o - \mu g t/V_\infty. \tag{24}$$

Further analysis then yields the coordinates of the aircraft in earth axes as follows

$$x = x_o + \frac{2V_\infty^2}{\mu g} \cos\left(\text{DIVE}_o - \frac{\mu g t}{2V_\infty}\right)\sin\left(\frac{\mu g t}{2V_\infty}\right) \tag{25(a)}$$

$$y = 0 \tag{25(b)}$$

$$z = z_o - \frac{2V_\infty^2}{\mu g} \sin\left(\text{DIVE}_o - \frac{\mu g t}{2V_\infty}\right)\sin\left(\frac{\mu g t}{2V_\infty}\right). \tag{25(c)}$$

When μ is small, equations (25) reduce to

$$x = x_o + V_\infty t \cos(\text{DIVE}_o) \tag{26(a)}$$

$$y = 0 \tag{26(b)}$$

$$z = z_o - V_\infty t \sin(\text{DIVE}_o). \tag{26(c)}$$

The case $\mu = 0$, of course, corresponds to a straight line trajectory.
From figure 1, it is clear that the components of aircraft acceleration in aircraft axes are constant and are given by

$$\ddot{x}_A = \mu g \sin(\text{ATTACK}) \tag{27(a)}$$

$$\ddot{y}_A = 0 \tag{27(b)}$$

$$\ddot{z}_A = -\mu g \cos(\text{ATTACK}). \tag{27(c)}$$

Also, the Euler angles describing the orientation of the aircraft with respect to earth axes are

$$\text{AROLL} = \pi \tag{28(a)}$$

$$\text{APITCH} = -(\text{DIVE-ATTACK}) \tag{28(b)}$$

and

$$AYAW = 0. \qquad (28(c))$$

Substitution of these angles into the direction cosine matrix C_A and application of the transformation $\ddot{\underline{x}}_E = C_A^{-1} \ddot{\underline{x}}_A$ gives the components of aircraft acceleration in earth axes, namely,

$$\ddot{x}_E = \mu g \sin(DIVE) \qquad (29(a))$$

$$\ddot{y}_E = 0 \qquad (29(b))$$

$$\ddot{z}_E = \mu g \cos(DIVE). \qquad (29(c))$$

2.5 Derivation of the equations of motion

We must now derive the six equations of motion which describe the motion of the bomb relative to the aircraft.
Let

(i) $\underline{X}$ be the position vector of the bomb and $\underline{X}^*$ be the position vector of the aircraft with respect to some axis system,

(ii) Subscripts E and A refer to earth and aircraft axes, and

(iii) C_A be the 3 x 3 direction-cosine matrix describing the orientation of the aircraft relative to earth axes.

Then

(i) $\underline{X}_E$ is the position vector of the bomb relative to earth axes,

(ii) $\underline{X}_E^*$ is the position vector of the aircraft relative to earth axes, and

(iii) $\underline{X}_A$ is the position vector of the bomb relative to aircraft axes.

Thus,

$$\underline{X}_A = C_A(\underline{X}_E - \underline{X}_E^*), \qquad (30)$$

and $\frac{d}{dt}$ of equation (30) gives

$$\dot{\underline{X}}_A = C_A(\dot{\underline{X}}_E - \dot{\underline{X}}_E^*) + \dot{C}_A(\underline{X}_E - \underline{X}_E^*). \qquad (31)$$

Equation (21) then yields

$$\dot{\underline{X}}_A = C_A(\dot{\underline{X}}_E - \dot{\underline{X}}_E^*) + \Omega_A C_A(\underline{X}_E - \underline{X}_E^*), \qquad (32)$$

where C_A is given by equation (4) and Ω_A is given by equation (15).
Now define

$$\underline{u} = C_A(\dot{\underline{X}}_E - \dot{\underline{X}}_E^*), \qquad (33)$$

where $\underline{u}$ would describe the velocity of the bomb centre of gravity with respect to aircraft axes if C_A were constant, that is, if the aircraft were not rotating about its own centre of gravity.
Then equations (33) and (30) in equation (32) give

$$\dot{\underline{X}}_A = \underline{u} + \Omega_A \underline{X}_A \tag{34}$$

and $\frac{d}{dt}$ of equation (33) yields

$$\dot{\underline{u}} = C_A \ddot{\underline{X}}_E - C_A \ddot{\underline{X}}^*_E + \Omega_A \underline{u}. \tag{35}$$

Now $\ddot{\underline{X}}_E$ is the acceleration of the bomb with respect to earth axes, expressed in earth axes. Thus, since earth axes represent an inertial system, we can write

$$\ddot{\underline{X}}_E = \underline{F}_E/m, \tag{36}$$

where $\underline{F}_E$ are the external forces on the bomb, expressed in earth axes, and m is the mass of the bomb.

Then, premultiplication of equation (36) by the 3 x 3 matrix C_A expresses the forces in aircraft axes; thus

$$C_A \ddot{\underline{X}}_E = C_A \underline{F}_E/m = \underline{F}_A/m. \tag{37}$$

Also, since $\ddot{\underline{X}}^*_E$ represents the acceleration of the aircraft relative to earth axes, then $C_A \ddot{\underline{X}}^*_E$ represents the same acceleration but now expressed in aircraft axes. Thus, by equations (27), we have

$$C_A \ddot{\underline{X}}^*_E = \begin{pmatrix} \mu g \sin(\mathrm{ATTACK}) \\ 0 \\ -\mu g \cos(\mathrm{ATTACK}) \end{pmatrix}. \tag{38}$$

Finally, substitution of equations (37) and (38) into equation (35) gives

$$\dot{\underline{u}} = \underline{F}_A/m - \begin{pmatrix} \mu g \sin(\mathrm{ATTACK}) \\ 0 \\ -\mu g \cos(\mathrm{ATTACK}) \end{pmatrix} + \Omega_A \underline{u}. \tag{39}$$

Equations (34) and (39) are thus the two first-order differential equations which can be integrated to give the motion of the bomb centre of gravity relative to aircraft axes.

To derive the equations of motion describing the attitude of the bomb relative to aircraft axes, we define $\underline{h}_B$ the angular momentum vector as

$$\underline{h}_B = \begin{pmatrix} p\ I_{XX} \\ q\ I_{YY} \\ r\ I_{ZZ} \end{pmatrix}, \tag{40}$$

where I_{XX}, I_{YY} and I_{ZZ} are the moments of inertia of the bomb about its own principal axes and p, q and r are the angular rates of the bomb about these same X, Y and Z axes respectively.

The inertia tensor is not a constant if it is calculated in the space coordinate system (earth axes) because, in this system, the values of the coordinates change with time. In the moving (bomb) system, however, the inertia tensor is constant. For this reason it is advantageous to work in the bomb system.

If $\underline{h}_E$ is the vector representing the angular momentum of the bomb in earth axes, then

$$\underline{h}_B = C_B\, \underline{h}_E, \tag{41}$$

where C_B is the direction-cosine matrix which describes the orientation of the bomb relative to earth axes.

$\frac{d}{dt}$ of equation (41) gives

$$\dot{\underline{h}}_B = C_B\, \dot{\underline{h}}_E + \dot{C}_B\, \underline{h}_E. \tag{42}$$

Then equations (21), (41) and (42) give

$$\dot{\underline{h}}_B = C_B\, \dot{\underline{h}}_E + \Omega_B\, \underline{h}_B. \tag{43}$$

Now, since earth axes represent an inertial system, we can write

$$\dot{\underline{h}}_E = \underline{M}_E, \tag{44}$$

where $\underline{M}_E$ is the vector representing external moments (torque) on the bomb, expressed in earth axes.

Thus, premultiplying equation (44) by C_B expresses the moments in bomb axes, namely

$$C_B\, \dot{\underline{h}}_E = \underline{M}_B. \tag{45}$$

Finally, equations (43) and (45) yield

$$\dot{\underline{h}}_B = \underline{M}_B + \Omega_B\, \underline{h}_B. \tag{46}$$

Equations (40) and (46) thus enable the calculation of the bomb angular rates p, q and r. In most cases, equations (10) would then give the Euler angular rates $\dot{\varphi}$, $\dot{\theta}$ and $\dot{\beta}$ which could then be integrated to give the Euler angles φ, θ and β as functions of time.

However, equations (10) have a singularity at $\theta = \pi/2$ and thus, for values of θ in the neighbourhood of $\pi/2$, truncation errors will occur. It is therefore safer to use, not the Euler angles themselves, but the direction-cosines of these angles as dependent variables.

Thus, let C be the direction-cosine matrix which represents the orientation of the bomb relative to the aircraft. If $\underline{b}$ is a vector fixed in bomb axes and $\underline{a}$ is a vector fixed in aircraft axes, then

$$\underline{b} = C\underline{a}. \tag{47}$$

Further, if $\underline{e}$ is a vector in the earth axis system, we have

$$\underline{b} = C_B\underline{e} \tag{48}$$

and

$$\underline{a} = C_A\underline{e}. \tag{49}$$

Simple manipulation then gives

$$CC_A = C_B \tag{50}$$

or

$$C = C_B C_A^{-1}. \tag{51}$$

$\frac{d}{dt}$ of equation (50) yields

$$\dot{C}C_A + C\dot{C}_A = \dot{C}_B. \tag{52}$$

Equation (21) applied to equation (52) thus gives

$$\dot{C}C_A + C\,\Omega_A\,C_A = \Omega_B\,C_B. \tag{53}$$

Post-multiplying equation (53) by C_A^{-1} and substituting equation (50) finally gives

$$\dot{C} = \Omega_B\,C - C\,\Omega_A. \tag{54}$$

The 3 x 3 matrix-equation (54) thus represents 9 first-order differential equations for the calculation of the elements of the orientation matrix C.

The procedure is therefore to

(i) Integrate equation (46) to calculate $\underline{h}_B$.

(ii) Use equation (40) to give p, q and r.

(iii) Substitute in equation (15) to give Ω_B.

(iv) Calculate Ω_A from the known aircraft angular rates p_A, q_A and r_A.

(v) Substitute into equation (54) and integrate it to yield the 3 x 3 matrix C.

If so desired, the Euler angles associated with C can be calculated by the following formulae, which are derived from equation (4):-

$$\theta = \tan^{-1}(C_{31}/\sqrt{C_{32}^2 + C_{33}^2}) \tag{55(a)}$$

$$\sin\beta = -C_{21}/\cos\theta \tag{55(b)}$$

$$\cos\beta = C_{11}/\cos\theta \tag{55(c)}$$

$$\sin\varphi = -C_{32}/\cos\theta \tag{55(d)}$$

$$\cos\varphi = C_{33}/\cos\theta \tag{55(e)}$$

Equation (55(a)) assumes that θ is acute.

However, as noted in the next section, it is usually more meaningful to use projected angles rather than Euler angles.

2.6 Transformations between Euler angles and projected angles

Euler angles are always difficult for the experimenter to visualize. For instance, in defining the orientation of the bomb relative to aircraft axes, it is extremely difficult to relate the bomb's attitude to the given Euler angles. It is therefore convenient to express the attitude in terms of projected angles.

Thus, let φ^*, θ^* and β^* be the projected roll, pitch and yaw of the bomb relative to aircraft axes, that is, the roll, pitch and yaw angles of the image of the bomb when projected upon the aircraft YZ, XZ and XY planes respectively.

To find projected pitch (θ^*) and projected yaw (β^*) it is necessary to find the coordinates of the unit vector in the direction of the bomb's nose (X axis) as represented in aircraft axes. This is given by

$$\begin{pmatrix} XN \\ YN \\ ZN \end{pmatrix} = C^{-1} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \tag{56}$$

where C is the matrix describing the orientation of the bomb relative to aircraft axes.

Since C is an orthogonal matrix, then equation (56) gives

$$XN = C_{11} \tag{57(a)}$$

$$YN = C_{12} \tag{57(b)}$$

$$ZN = C_{13}. \tag{57(c)}$$

Thus

$$\theta^* = \tan^{-1}(-ZN/XN) = -\tan^{-1}(C_{13}/C_{11}) \tag{58}$$

and

$$\beta^* = \tan^{-1}(YN/XN) = \tan^{-1}(C_{12}/C_{11}). \qquad (59)$$

Similarly the projected roll (φ^*) can be found from the coordinates of the unit vector in the direction of the bomb's reference fin (Z axis) as represented in aircraft axes, given by

$$\begin{pmatrix} XF \\ YF \\ ZF \end{pmatrix} = C^{-1}\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} C_{31} \\ C_{32} \\ C_{33} \end{pmatrix} = \begin{pmatrix} \sin\theta \\ -\sin\varphi\cos\theta \\ \cos\varphi\cos\theta \end{pmatrix}.$$

Since roll is always the first of the Euler angles, projected roll (φ^*) and Euler roll (φ) are equal, giving

$$\sin\varphi^* = -C_{32}/\cos\theta \qquad (60(a))$$

and

$$\cos\varphi^* = C_{33}/\cos\theta \qquad (60(b))$$

Differentiation of equations (60) with respect to time shows that

$$\dot{\varphi}^* \equiv \dot{\varphi} = (C_{33}\,\dot{C}_{32} - \dot{C}_{32}\,C_{33})/(C_{32}^2 + C_{33}^2). \qquad (61)$$

Projected pitch and yaw rates are obtained by differentiating equations (58) and (59) to give

$$\dot{\theta}^* = (C_{11}\,\dot{C}_{13} - \dot{C}_{13}\,C_{11})/(C_{11}^2 + C_{13}^2) \qquad (62)$$

and

$$\dot{\beta}^* = (C_{12}\,\dot{C}_{11} - \dot{C}_{11}\,C_{12})/(C_{11}^2 + C_{12}^2). \qquad (63)$$

Further, equation (4) can be used to substitute for the elements of C to give the projected angles in terms of the Euler angles, namely:-

$$\varphi^* \equiv \varphi \qquad (64(a))$$

$$\tan\theta^* = \tan\theta\cos\varphi - \sin\varphi\tan\beta/\cos\theta \qquad (64(b))$$

and

$$\tan\beta^* = \tan\theta\sin\varphi + \cos\varphi\tan\beta/\cos\theta \qquad (64(c))$$

The inverses of equations (64) are then

$$\varphi \equiv \varphi^* \qquad (65(a))$$

$$\tan\theta = \tan\theta^* \cos\varphi^* + \tan\beta^* \sin\varphi^* \tag{65(b)}$$

and

$$\tan\beta = \cos\theta(\tan\beta^* \cos\varphi^* - \tan\theta^* \sin\varphi^*). \tag{65(c)}$$

2.7 Relation between Projected Angular Rates and Bomb Angular Rates

In Section 2.2, p, q and r were defined as the angular rates of rotation, relative to fixed axes, of the rotating system about its own X, Y and Z axes respectively. Thus we here define p, q and r to be the angular rates of the bomb, relative to earth axes, about the bomb's X, Y and Z axes respectively.

If p_A, q_A and r_A are the angular rates of rotation of the aircraft, relative to earth axes, about the aircraft's X, Y and Z axes respectively, then we can write

$$\underline{p} = C\underline{p}_A + d\underline{p}, \tag{66}$$

where

$$\underline{p} = \begin{pmatrix} p \\ q \\ r \end{pmatrix}$$

$$\underline{p}_A = \begin{pmatrix} p_A \\ q_A \\ r_A \end{pmatrix},$$

C is the direction cosine matrix describing the orientation of the bomb relative to the aircraft and $d\underline{p}$ is the effect of the rates of change of the Euler angles of C.

Now, differentiating equations (65) with respect to time gives

$$\dot{\varphi} \equiv \dot{\varphi}^* \tag{67(a)}$$

$$\sec^2\theta\,\dot{\theta} = \sec^2\theta^* \cos\varphi^*\,\dot{\theta}^* + \sec^2\beta^* \sin\varphi^*\,\dot{\beta}^* + \tan\beta\,\dot{\varphi}^*/\cos\theta \tag{67(b)}$$

and

$$\sec^2\beta\,\dot{\beta} = \cos\theta(\sec^2\beta^* \cos\varphi^*\,\dot{\beta}^* - \sec^2\theta^* \sin\varphi^*\,\dot{\theta}^* - \tan\theta\,\dot{\varphi}^*) - \tan\theta\tan\beta\,\dot{\theta} \tag{67(c)}$$

Then, equations (4), (9) and (67) give

$$dp = C_{21}(C_{32} \sec^2 \beta^* \dot{\beta}^* - C_{33} \sec^2 \theta^* \dot{\theta}^*) + \cos\theta\, \dot{\varphi}^*/\cos\beta \qquad (68(a))$$

$$dq = C_{11}(C_{33} \sec^2 \theta^* \dot{\theta}^* - C_{32} \sec^2 \beta^* \dot{\beta}^*) \qquad (68(b))$$

and

$$dr = C_{11}(C_{22} \sec^2 \beta^* \dot{\beta}^* - C_{23} \sec^2\theta^* \dot{\theta}^*). \qquad (68(c))$$

Equations (66) and (68) therefore provide a means of calculating bomb angular rates from projected angular rates, the latter being specified by the experimenter prior to the running of the Captive Trajectory Yawmeter System.

3. DETERMINATION OF THE FORCES AND MOMENTS

The Captive Trajectory Yawmeter System determines the aerodynamic forces and moments on the bomb by

(a) selecting the most appropriate result from data which have been previously acquired on an approximately 1/8th scale model of the store in uniform flow, and

(b) calculating increments to the store loading due to the known non-uniform flow, as measured on-line by a yawmeter probe.

3.1 Store characteristics in uniform flow

Tabulated data representing the characteristics of the store in a uniform flow are required for this procedure. The tables reside on disk and consist of ten coefficients as a function of reference Mach number, total pitch and total roll.

The yawmeter probe is programmed to take readings at twelve stations along the calculated position of the store centre-line, including three stations located at the leading edge, mid-point and trailing edge of the mean aerodynamic chord of the fins.

A reference point, usually the station at the mid-point of the fins, is selected and, from these reference flow conditions, a Mach number, total pitch and total roll are calculated. These quantities are then used to obtain the ten uniform-flow coefficients from the disk.

In calculating the incidence and sideslip upon which total pitch and total roll are based, some allowance must be made for the curvature of the stream past the fins. Here use is made of thin aerofoil theory from which it can be shown that "the lift of an aerofoil of camber γ and incidence α is equal to the lift of a straight aerofoil at incidence $(\alpha + 2\gamma)$". This result is easily proved using page 91 of reference 6.

Consider the velocities at the three fin stations in the XZ plane of figure 3.

The slope of the streamline at any station is given by

$$\frac{dz}{dx} = \dot{z}/\dot{x} = W/U, \qquad (69)$$

where U and W are the measured flow velocity components in the X and Z directions respectively. A parabola is fitted through the streamline slopes at the three fin stations and the equation of the streamline is thus obtained by integration of equation (69) with respect to x.

The equation of the chord (figure 3) is given by

$$z(x) = (z_c/x_c)\ x. \tag{70}$$

Thus the mean height is given by

$$h = \Delta/\cos\alpha = (z_B - z_c\ x_B/x_c)/\cos\alpha \tag{71}$$

The mean chord, quite clearly, is given by

$$K = x_c/\cos\alpha \tag{72}$$

Thus, the camber is given by

$$\gamma \equiv h/K = (z_B - z_c\ x_B/x_c)/x_c. \tag{73}$$

In addition, the mean incidence is given by

$$\alpha = \tan^{-1}(z_c/x_c). \tag{74}$$

Equations (73) and (74) thus provide the means for calculating the effective incidence $\alpha_e = (\alpha + 2\gamma)$.

The above procedure is repeated in the XY plane to determine the effective sideslip $\beta_e = (\beta + 2\gamma)$.

In deriving the most appropriate uniform flow coefficients from the disk data store, it is assumed that the tail loading contained in these coefficients is representative of the tail loading in the curved flow field. The data store coefficients are, of course, based on free stream dynamic pressure. Thus, to obtain the coefficients appropriate to the reference point, the data store coefficients are multiplied by reference point dynamic pressure and divided by free stream dynamic pressure.

3.2 Corrections for non-uniform flow

Since the interpolated coefficients of the previous section are for a uniform flow, the flow conditions implied at each store measurement station are equal to those at the store reference point. However, corrections must be applied to allow for changes in body loading forward of the fins, due to the differences between the measured flow conditions at the store stations and the uniform flow implied by the interpolated coefficients.

Allen and Perkins(ref.7) derive an expression similar to that derived by Munk for the potential cross force on slender bodies, given by

$$f = q\frac{dS}{dx}\sin 2\alpha, \tag{75}$$

where q is the dynamic pressure, and α is body-axis incidence relative to the free-stream flow direction.

They also note that, from the work of Ward(ref.8), it can be shown that the potential cross-force is directed midway between the normal to the axis of revolution and the normal to the wind direction.

This applied to equation (75) gives a cross-force of

$$Z_P = q_\infty \frac{dS}{dx} \sin 2\alpha \cos \tfrac{1}{2}\alpha \tag{76(a)}$$

and

$$Y_P = q_\infty \frac{dS}{dx} \sin 2\beta \cos \tfrac{1}{2}\beta \tag{76(b)}$$

for the Z and Y potential flow contributions respectively.

A second term is due to viscosity and can be written as

$$f_V = \eta\, C_{D_C}\, D\, q \sin^2 \theta_T, \tag{77}$$

where C_{D_C} is the cross-flow drag coefficient,

D is the cross-sectional diameter of the bomb (i.e. $S = \pi D^2/4$),

η is the cross-flow drag proportionality factor,

θ_T is the total pitch of the bomb (see figure 4),

and q is dynamic pressure $(=\tfrac{1}{2}\rho\,|\underline{V}|^2)$.

η arises from the fact that the ratio of the cross-flow drag for a finite length cylinder to that for an infinite length cylinder is less than one (see reference 9).

Now, since f_V acts in the total Z axis negative direction, the components of the viscous force in the Z and Y directions will be given by (see figure 4)

$$Z_V = -\tfrac{1}{2}\rho\,\eta\, C_{D_C}\, D\, |\underline{V}|^2 \sin^2 \theta_T \cos \varphi_T \tag{78(a)}$$

and

$$Y_V = -\tfrac{1}{2}\rho\,\eta\, C_{D_C}\, D\, |\underline{V}|^2 \sin^2 \theta_T \sin \varphi_T. \tag{78(b)}$$

But, as shown in figure 4,

$$U = -|\underline{V}| \cos \theta_T, \tag{79(a)}$$

$$V = -|\underline{V}| \sin \theta_T \sin \varphi_T \tag{79(b)}$$

and

$$W = -|\underline{V}| \sin \theta_T \cos \varphi_T. \tag{79(c)}$$

Thus, equations (79) in equations (78) yield

$$Z_V = \tfrac{1}{2} \rho \eta C_{D_C} D W \sqrt{V^2 + W^2} \qquad (80(a))$$

and

$$Y_V = \tfrac{1}{2} \rho \eta C_{D_C} D V \sqrt{V^2 + W^2} . \qquad (80(b))$$

Finally, as noted in reference 10, allowance must be made for buoyancy. This results in the components

$$Z_B = -V_\infty \rho S \frac{dW}{dx} \qquad (81(a))$$

and

$$Y_B = -V_\infty \rho S \frac{dV}{dx} \qquad (81(b))$$

where V_∞ is free-stream velocity and ρ is local air density. The buoyancy term arises because the streamlines at any point are curved. This is confirmed by measurements taken by the yawmeter probe along the calculated position of the store centre-line. However, buoyancy is not expected to contribute a large amount to the overall cross-force.

The Captive Trajectory Yawmeter System thus defines total cross-forces of

$$Z = Z_P + Z_V + Z_B \qquad (82(a))$$

and

$$Y = Y_P + Y_V + Y_B . \qquad (82(b))$$

Uniform-flow theory says that flow conditions at all body stations are equal to those at the reference point. Thus, the calculations implied by equations (76) to (82) are repeated, with only the body geometry being allowed to vary between stations, to give the local cross-force predicted at each station by uniform flow theory, namely Z_u.

Then the increment in cross-force due to the non-uniformity of the flow at each station is given by

$$\Delta Z = Z - Z_u .$$

Similarly, a pitching moment at each station can be calculated, with resp[illegible] to an origin at the centre of gravity of the store, as

$$M = Z(x - x_{CG}) .$$

Thus the increment due to non-uniformity of the flow is given by

$$\Delta M = (Z - Z_u)(x - x_{CG}).$$

Finally, this analysis can be applied to the yaw plane to yield values for ΔY and yawing moment ΔN.

In the limit, as the flow becomes uniform, these corrections all tend to zero.

4. SYSTEM DESCRIPTION

The software for the Captive Trajectory Yawmeter System consists of an overlay system comprising a resident program plus several links which overlay each other in core. This has been necessary because of the limited core size (24K 18-bit words) of the UNICHANNEL-15 computer.

All relevant variables have been allocated to Blank COMMON, and this resident COMMON area is written on disk after the calculation of each trajectory point. This feature allows the trajectory calculations to be resumed from a previous trajectory and thus provides a useful restart facility.

At the beginning of a new trajectory, the COMMON area is initiated by reading-in a Constants Block from the disk. The Constants Block is set-up by means of a special Editor prior to the running of an experiment and resides unchanged on disk until it is required to change any of the parameters. This removes the need to type-in a large amount of information at the beginning of each run.

The Constants Block includes the initial position, attitude and rates of the store because the present CTYS begins its calculations after the ejector has operated and thus the effects of the ejector must already be present in the Constants Block when the CTYS begins its operation. However, the ejector phase of the CTYS has been developed separately(ref.4) and work is currently underway to incorporate it as an integral part of the CTYS.

4.1 Operation of the Constants Editor

The Constants Editor (EDCON) either sets up a new file or allows modifications to be carried out on an existing file. It operates in an interactive manner by prompting the operator and acquiring answers to the questions it asks.

Listings of EDCON and its utility routines DECDE and IDECDE are given in Appendix I. These listings specify the elements of the arrays T and IT, which together comprise the COMMON block for the CTYS.

The loading sequence for EDCON is:

```
$A RKA -5
$GLOAD
 LOADER V3A000
>←EDCON <ALTMODE>
```

When EDCON has loaded, it announces its presence and asks the operator to specify a file name. This file name will denote either a new file to be created or an existing file to be modified.

If the file is new, EDCON enters an Input mode and acquires the values of all relevant constants by means of a series of questions and answers. Each prompt by EDCON includes the units in which the answer is expected, thus providing a convenient way to permanently establish the units of each input.

If the file is not new, EDCON enters an Edit mode which first checks for the existence of the file and, if found, opens it. It then prompts the operator with

ALTER PARAMETERS?

A negative answer (N) at this stage would simply produce a listing of the unmodified Constants Block. However, an affirmative (Y) would cause a second prompt asking the operator to enter N, where N denotes the following:

(1) Bomb parameters

(2) Aircraft parameters

(3) Probe parameters

(4) Miscellaneous information

(5) Measurement Stations

(6) Diameters at Measurement Stations.

The value for N directs EDCON to the appropriate area in the Constants Block, where it begins to list each parameter (just as it did in Input mode) together with the current value of that parameter. If the operator wishes to change a parameter, he types in the new value. A CARRIAGE RETURN will accept the old value.

The operator may allow listing to continue to the end of the section, changing the values of any desired parameters, or he may terminate this phase by typing CONTROL P. In either event, EDCON will then again type

ALTER PARAMETERS?

The process is repeated until the operator finally answers N(0) to this question. Then EDCON

(i) calculates the aircraft Euler angles and their rates, relative to earth axes,

(ii) calculates the Euler angles defining the orientation of the bomb relative to aircraft axes,

(iii) determines the bomb's maximum cross-sectional area, and its corresponding diameter,

(iv) initializes the direction-cosine matrix defining the orientation of the bomb relative to aircraft axes, according to equation (4),

(v) calculates the angular rates of the bomb as described in Section 2.7, then

(vi) calculates the initial values of bomb angular momentum as defined by equation (40),

(vii) finds the linear accelerations of the aircraft relative to aircraft axes, as defined by equations (27), and finally

(viii) calculates the free-stream values for velocity, velocity of sound and ambient temperature, from free-stream Mach number and the height of the aircraft.

EDCON then writes the file on disk and prints the values of all entered constants on the line printer.

The process of entering the constants is straightforward and should not cause any difficulty. However, it should be remembered that the roll, pitch and yaw of the bomb, together with their rates, are entered as projected angles, not Euler angles.

4.2 Operation of the Main Program

The Main Program (MAIN), like the COMMON area, is resident in core throughout the entire trajectory.

Upon loading, its first task is to initiate the traverse rig console button. This primes the program so that it will change to a "bomb-alone" trajectory when the button is pressed, namely, when the operator judges that the bomb is sufficiently clear of the aircraft to be in a nominally uniform flow. The probe is then no longer required and a trajectory to impact is calculated.

The Main Program next initiates the system for Direct-Access Input/Output, which permits the user to reference directly any record in a file without indexing from the file's beginning up to the desired record. This feature is, of course, essential to an operation such as the CTYS, where very large numbers of records must be accessed in a random manner.

After announcing its presence, MAIN prompts the operator to determine whether the trajectory is new or is simply the continuation of a previous one. If the trajectory is not new, the system reads into core the contents of file 'TFILE CTS', the last image of the COMMON area calculated during the previous trajectory. This simple restart facility thus allows the continuation of a trajectory which has been interrupted by a system failure or for some other reason.

If the trajectory is in fact new, the operator is asked two more questions:

(i) MAIN asks whether the store under consideration is MK82 or Karinga. This conditions the system to take data from the appropriate data store.

(ii) The operator is then asked to enter a file name, which specifies the name of the Constants Block set up by EDCON, as described in Section 4.1.

MAIN then starts the "flight". Subroutine DAUX is called to initialize derivatives after which the dive angle is calculated by means of equation (24) and the aircraft orientation relative to earth axes is determined using equations (28).

At this point in the calculation of each trajectory point, MAIN tests the parameter IFREE, which is set positive when the traverse rig console button is pressed to signify a transition to a "bomb-alone" trajectory. If IFREE has become positive, a once-only call is made to subroutine TRFREE, which converts bomb position from aircraft axes to earth axes, the latter being more meaningful to the calculation of trajectory to impact.

For all trajectory points, MAIN then calculates, for the current height, the following quantities:

$$p = 10335.11\, g(1 - 2.2559 \times 10^{-5}\, h)^{5.256103},$$

$$T = 14.99 - 0.0065\, h,$$

$$a_\infty = \sqrt{401.742T + 109809.48}\ ,$$

$$V_\infty = a_\infty M_\infty$$

and

$$q = \tfrac{1}{2} \gamma M_\infty^2\, p,$$

where g is gravitational acceleration (m/s^2),

h is height (m),

p is static pressure ($newton/m^2$),

T is ambient temperature (^oC),

a_∞ is velocity of sound (m/s),

V_∞ is free-stream velocity (m/s),

M_∞ is free-stream Mach number,

q is dynamic pressure ($newton/m^2$) and

γ is the ratio of specific heats.

After expressing the gravitational force on the bomb in aircraft axes, MAIN then takes one of two courses of action, depending on whether the bomb-alone phase is in operation.

(a) If the store is still in the vicinity of the aircraft,

(i) the flow properties along the centre-line of the aircraft, are determined (CALL FLOW),

(ii) the effective incidence and sideslip are calculated using the methods of Section 3.1 (CALL CORECT),

(iii) the force and moment coefficients due to uniform flow are interpolated from the disk data store (CALL CREF),

(iv) the increments to the force and moment coefficients arising from the non-uniform flow are calculated (CALL CINC), and finally,

(v) the forces and moments on the store from all sources are determined, the forces being in aircraft axes and the moments being in bomb axes.

(b) If the store has entered the "bomb-alone" phase of its trajectory, the wind velocity components in bomb axes are just the negatives of the bomb velocity components in earth axes. The values are then used in steps (a) (iii) and (a) (v) above to determine the forces and moments on the store.

Thus, to calculate the next trajectory point, it only remains to invoke a fourth-order Runge Kutta integration and print out the result.

However, the steps described in (a) and (b) above are carried out only once at each trajectory point, whereas the Runge Kutta process requires values of the forces and moments at several places within the time interval under consideration. If this condition is not met, an oversimplification of the integration process occurs and an instability is produced in the solution for roll, pitch and yaw.

It would be quite impractical to calculate forces and moments at every point in the time interval required by the Runge Kutta process, because of the large amounts of processing time involved. Thus, to overcome the instability, a time-wise parabola is fitted to the three most recent values for each of the three forces and each of the three moments. The Runge Kutta process then uses each parabola (in Subroutine DAUX) to extrapolate into the current time interval and thus to produce a more accurate estimate of each force and moment.

The process described above is repeated to calculate each trajectory point, the calculations continuing until either the maximum time has been exceeded or the store has struck the ground.

Specific details of the subroutines used by the Captive Trajectory Yawmeter System are given in subsequent sections. A listing of the Main Program appears in Appendix II.

4.3 Subroutines TRSET and TRFREE

TRSET is a small routine, coded in Assembly Language (MACRO-15), which initiates the Skip Chain (see reference 11) to receive interrupts from the traverse rig console button. It is called by MAIN when MAIN is first loaded and sets the parameter IFREE to zero.

When the traverse rig console button is pressed, control is transferred to another part of TRSET, which clears the flag, turns on the Program Interrupt, sets IFREE to +1 and then returns control to the interrupt address.

At the beginning of each trajectory point, MAIN tests IFREE to see if it has been set positive (i.e., to +1). If it has, then MAIN calls Subroutine TRFREE in order to perform the transformation to a "bomb-alone" trajectory.

TRFREE first sets the parameter FREE to +1. It then performs the transformation of coordinates as follows.

If $\underline{a}$, $\underline{b}$ and $\underline{c}$ are vectors fixed in aircraft, bomb and earth axes respectively, then

$$\underline{a} = C_A \underline{e} \tag{83}$$

$$\underline{b} = C_B \underline{e} \tag{84}$$

and

$$\underline{b} = C \underline{a} , \tag{85}$$

where C_A, C_B and C are direction-cosine matrices as defined in equation (4).

Equations (83), (84) and (85) then give

$$C_B = C\, C_A . \tag{86}$$

Since the orientation of the bomb must now be referred to earth axes instead of aircraft axes, the matrix C is replaced by the matrix C_B, as defined in equation (86).

Similarly, if $\underline{x} = (x,y,z)$ is the vector which, until the transition, has represented the position of the bomb centre of gravity relative to earth axes, then this can be expressed as

$$\underline{x} = C_A^{-1} \underline{x} + \underline{x}_A , \tag{87}$$

where $\underline{x}_A$ is the vector which represents the position of the aircraft centre of gravity relative to earth axes. The components of $\underline{x}_A$ are given in equations (25) and (26). The vector $\underline{x}$ thence describes the position of the bomb centre of gravity relative to earth axes.

Differentiation of equation (87) with respect to time gives

$$\underline{u} = C_A^{-1} \underline{u} + \begin{pmatrix} V_\infty \cos(\mathrm{DIVE}) \\ 0 \\ -V_\infty \sin(\mathrm{DIVE}) \end{pmatrix} \tag{88}$$

Now $\underline{u}$ signifies the velocity of the bomb centre of gravity relative to earth axes rather than relative to aircraft axes.

Finally, TRFREE calls Subroutine DAUX (to set up the new derivatives), sets the gravitational acceleration components (which are then constant) to earth axes, and then prints a message on the line printer to signify that the "bomb-alone" trajectory has begun.
Listings of TRSET and TRFREE are given in Appendix III.

4.4 General purpose routines (ARCTAN, TRANS, CTRANS and PARAB)

Functions ARCTAN (Y,X) calculates the angle θ whose sine is given by Y and whose cosine is given by X. θ can range from 0 to 2π.
TRANS and CTRANS are entry points to a single FORTRAN routine. For given inputs ROLL, PITCH, YAW (Euler angles), INV and $\underline{x}$, TRANS performs one of the coordinate transformations

$$\underline{x}^1 = C\underline{x} \qquad (\text{INV} = +1) \tag{89(a)}$$

or

$$\underline{x}^1 = C^{-1}\,\underline{x} \qquad (\text{INV} = -1), \tag{89(b)}$$

where C is calculated according to equation (4) and is the direction-cosine (orientation) matrix associated with the transformation.
CTRANS is a special case of TRANS in that it transforms specifically from aircraft to bomb axes (INV = +1) or vice versa (INV = -1). Since the nine elements of C in this case are all dependent variables of the time integration, these elements are always available for use in equations (89) and thus do not have to be calculated.
Subroutine PARAB simply fits a parabola

$$f(x) = a + b(x - x_A) + c(x - x_A)^2$$

to three values of the function f(x) and stores the coefficients in the array ABC.
A listing of Subroutines ARCTAN, TRANS, CTRANS and PARAB is given in Appendix IV.

4.5 The flow measurement and probe traverse routines

Subroutine FLOW, together with its associated subroutines CALC, TRAV, TRWAIT, ADRD and PROBE, is responsible for finding the flow properties in bomb axes at the twelve stations along the store centre-line.
FLOW first calls Subroutine CALC which

(i) transforms the coordinates of the first station from bomb axes to aircraft axes,

(ii) Converts the coordinates from metres to inches,

(iii) divides by the scale factor (SCALE) to convert from full scale to model scale length measurements,

(iv) transforms from aircraft axes to traverse rig axes, and

(v) calls Subroutine TRAV to move the yawmeter probe to the position calculated.

TRAV is an Assembly Language routine which drives the traverse rig to the wind tunnel coordinates (X,Y,Z) in the minimum time. It uses one hardware register to drive three coordinate axes simultaneously thus making its operation somewhat complicated. A full description of TRAV can be found in Section 3.8.2 of reference 5.
When the probe has arrived at the desired position, a probe settling delay of six seconds is allowed (by TRWAIT) and then the necessary pressure measurements are taken (by ADRD). Manifold, pitot and the two differential

pressures (DP13 and DP24) are acquired through a Raytheon Multiverter (multiplexed analog-to-digital converter), while the total pressure is acquired from a digital read-out of the S1 wind tunnel Boulton-Paul manometers.

Having acquired the pressure data, FLOW again calls CALC in order to start the probe moving towards the next measurement station. Thus, the probe movement to the next station and the processing of the measurements of the current station can proceed in parallel, so that delays can be kept to a minimum.

To calculate the flow properties at the current station from the pressure measurements, FLOW calls Subroutine PROBE, which calculates the ratio of pitot pressure to manifold pressure and prints an error message if this ratio is less than one.

PROBE uses the pressure measurements from the yawmeter probe to calculate a downwash and sidewash. This is accomplished by means of three probe calibration data stores which reside on disk. The three files are as follows:

(a) TMACHT PRB resides on logical unit 14 (.DAT 16_8) and is a table of Mach number as a function of pitch (PITCH) and the pitot-to-manifold pressure ratio (PRATIO), PITCH ranging from 0° to 29° in steps of 1° and PRATIO ranging from 1 to 4.2 in steps of 0.05. In the tables, PRATIO ranges more rapidly than PITCH; that is, the first 65 table entries are for a pitch of 0°, the next for a pitch of 1°, and so on up to the last 65 values which are for a pitch of 29°. In general, the record number corresponding to a particular PITCH and PRATIO is given by

$$R = 65.PITCH + 20(PRATIO-1)+1.$$

(b) PRATIO PRB is on logical unit 3 and is a table of pitot-to-total pressure ratio (RATIO) as a function of pitch ranging from 0° to 29° in steps of 1° and Mach number ranging from 0.4 to 1.4 in steps of 0.1. Again, Mach number ranges more rapidly than pitch. In general, the record number corresponding to a particular PITCH and MACH number is given by

$$R = 11.PITCH + 10.MACH-3.$$

(c) MACHCX PRB resides on logical unit 12 (.DAT 14_8) and is a table of downwash angle and sidewash angle as functions of the differential pressure coefficients (DC13, DC24) and Mach number. DC13 and DC24 each range from -2 to +2 in steps of 0.1 and Mach number ranges from 0.4 to 1.4 in steps of 0.1. Mach number ranges most slowly, then DC24 and finally DC13 ranges most rapidly. Thus, the first 41 entries each give a downwash and sidewash for a Mach number of 0.4 and a DC24 of -2, the next 41 entries correspond to a Mach number of 0.4 and a DC24 of -1.9, and so on up to the last 41 entries, which correspond to a Mach number of 1.4 and a DC24 of +2. In general, the record number corresponding to a particular DC13, DC24 and MACH number is given by

$$R = 41^2(10.MACH-3) + 41[10(DC24+2)] + 10(DC23+2)+1.$$

Returning to the operation of Subroutine PROBE, a pitot-to-manifold pressure ratio is calculated, a pitch equal to the last pitch calculated (obviously zero for the first time through) is assumed, and the file TMACHT (as described in (a) above) is accessed to give a corresponding Mach number (M).

After checking that the Mach number is in the range 0.4 to 1.4, PROBE then accesses PRATIO PRB (as in (b) above) to give a pitot-to-total pressure ratio for the assumed pitch and the now known value of Mach number. This ratio, together with the measured pitot pressure, then gives an estimate of total pressure, from which the static pressure is calculated from the formula

$$p_{static} = p_{total}/(1 + (\gamma - 1)M^2/2)^{\gamma/(\gamma-1)}.$$

Dynamic pressure (q) is then given by the relation

$$q = \tfrac{1}{2}\gamma\, p_{static}\, M^2.$$

The calculated value of q, together with the measured differential pressures DP13 and DP24, enables the differential pressure coefficients to be calculated from the formulae

$$DC13 = DP13/q$$

and

$$DC24 = DP24/q.$$

The file MACHCX PRB (as in (c) above) is then used to give values of downwash angle and sidewash angle as functions of DC13, DC24 and M.

The downwash and sidewash then enable a new estimate of pitch to be calculated and the process is then repeated using this updated value of pitch. The iteration is said to converge when successive approximations for pitch lie within 0.15° of each other but, in any event, an error message will be printed and execution terminated if more than 5 iterations are required.

All disk accesses are performed by subroutine READIN, which reads from the tables entries on either side of given values of the two independent variables and then forms a two-dimensional linear interpolation.

As with the data bank of store coefficients, the accesses to the disk to obtain probe calibrations are carried out by means of Direct-Access Input-Output commands, which permit the user to directly reference any record in a file without indexing from the file's beginning up to the desired record. Therefore, all calls to READIN must include, not only the logical unit number to define the file to be accessed, but also a record number which is calculated from the values of the independent variables under consideration. PROBE thus contains the necessary algorithms for computing appropriate record numbers associated with the files TMACHT, PRATIO and MACHCX.

When PROBE has calculated, relative to probe axes, the downwash angle (DOWNP) and sidewash angle (SIDEP) for the current store measurement station, control is then returned to Subroutine FLOW.

FLOW then uses the Mach number calculated by PROBE to determine the local velocity and then the components of local velocity relative to probe axes. These are given by

$$(a/a_\infty)^2 = (1 + \frac{\gamma - 1}{2} M_\infty^2)/(1 + \frac{\gamma - 1}{2} M^2) \tag{90}$$

$$|\underline{V}| = Ma \tag{91}$$

$$U_P = - |\underline{V}| / \sqrt{1 + \tan^2(\text{DOWNP}) + \tan^2(\text{SIDEP})} \tag{92(a)}$$

$$V_P = U_P \tan(\text{SIDEP}) \tag{92(b)}$$

and

$$W_P = U_P \tan(\text{DOWNP}), \tag{92(c)}$$

where a is the local velocity of sound and the subscript ∞ refers to free-stream values.

$\underline{V}$ is the wind vector and is shown in figure 5 together with the probe axes and the corresponding velocity components U_P, V_P and W_P.

The vector (U_P, V_P, W_P) is transformed, through the Euler angles probe roll and probe pitch, to wind tunnel axes, then to aircraft axes and finally to bomb axes. The final velocity components (UMS, VMS, WMS) returned by FLOW therefore represent the components of wind velocity relative to bomb axes.

FLOW repeats the entire process for each of the twelve measurement stations. To save time, the measurement stations are visited in the sequence (1,2,3, ..., 12) for odd trajectory points and in the sequence (12,11, ..., 1) for even trajectory points - a "zig-zag" effect.

Listings of Subroutines FLOW, CALC, TRAV, TRWAIT, ADRD, PROBE and READIN are given in Appendix V.

4.6 The uniform flow routines CORECT and CREF

Section 3.1 describes how uniform flow conditions are calculated at a reference point.

Subroutine CORECT uses the methods of Section 3.1 to calculate the effective incidence (α_e) and the effective sideslip (β_e), given by

$$\alpha_e = \alpha + 2\gamma \tag{93}$$

and

$$\beta_e = \beta + 2\gamma, \tag{94}$$

where α, β and γ are, respectively, the incidence, sideslip and camber defined in Section 3.1.

CORECT then returns control to the Main Program, which calculates total pitch (θ_T) and total roll (φ_T) according to the formulae

$$\theta_T = \tan^{-1}(\sqrt{(V/U)^2 + (W/U)^2}) \tag{95}$$

and

$$\varphi_T = \text{ARCTAN}(V/U, W/U). \tag{96}$$

Equations (95) and (96) derive from the equations (see figure 4)

$$U = -|\underline{V}| \cos \theta_T \quad (97(a))$$

$$V = -|\underline{V}| \sin \theta_T \sin \varphi_T \quad (97(b))$$

and

$$W = -|\underline{V}| \sin \theta_T \cos \varphi_T, \quad (97(c))$$

where $\underline{V}$ is the wind vector and U,V,W are the components of the wind vector relative to bomb axes.

The next step is to call subroutine CREF, which uses total pitch and total roll to read the appropriate force and moment coefficients from logical unit 13 (.DAT 15_8) of the disk.

This uniform flow data store is named 'MK82CX STC' for the Mk82 Bomb data and is named 'KARCX STC' for the Karinga store data. Each file consists of 8 x 13 x 16 = 1664 records, each record containing the 10 coefficients CX, CY, CZ, CL, CM, CN, CLP, CMQ, CNP and CYP for given Mach number, Total Pitch and Total Roll. The last four coefficients are dynamic derivatives and are given by

$$CLP = \frac{2V}{D}\frac{\partial CL}{\partial p} \quad \text{(Roll Damping)} \quad (98(a))$$

$$CMQ = \frac{2V}{D}\frac{\partial CM}{\partial q} \quad \text{(Pitch Damping)} \quad (98(b))$$

$$CNP = \frac{2V}{D}\frac{\partial CN}{\partial p} \quad \text{(Magnus Moment)} \quad (98(c))$$

and

$$CYP = \frac{2V}{D}\frac{\partial CY}{\partial p} \quad \text{(Magnus Force)} \quad (98(d))$$

where D is maximum cross-sectional diameter, V is local velocity and p, q, r are bomb Euler rates relative to total axes.

The range of Mach numbers is 0.4, 0.5, 0.6, 0.7, 0.8, 0.85, 0.9 and 0.95 (this range is currently being extended to Mach 1.2). Total roll ranges from -45° to $+45^\circ$ in steps of 7.5° while total pitch ranges from 0° to 30° in steps of 2°.

Mach number ranges most slowly, then total roll, and finally total pitch ranges the most rapidly. Thus, the first 16 entries correspond to a Mach number of 0.4 and a total roll of -45°, the next 16 to M = 0.4 and total roll = -37.5°, and so on up to the last 16 values which correspond to a Mach number of 0.9 and a total roll of $+45^\circ$.

In general, the record number corresponding to a particular Mach number, total roll (φ_T) and total pitch (θ_T) is given by

$$R = 208(M - 1) + 16[(\varphi_T + 45)/7.5] + [\theta_T/2] + 1,$$

where [ ] denotes the integer part and M is defined by

$$M = [10.MACH - 3] \text{ for } 0.4 \leqslant MACH \leqslant 0.8$$

and

$$M = [20.MACH - 11] \text{ for } 0.8 \leqslant MACH \leqslant 0.95.$$

CREF calls READIN twice to determine the flow values at the Mach number table entries above and below the required Mach number. It then interpolates over Mach number to give the final values of the coefficients. The total interpolation therefore consists of two two-dimensional interpolations over pitch and roll (one for each call to READIN) plus a one-dimensional interpolation over Mach number.

Allowing for the dynamic derivatives, CREF then calculates the coefficients relative to total axes as

$$TCX = CX \tag{99(a)}$$

$$TCY = CY + p\frac{\partial CY}{\partial p} \tag{99(b)}$$

$$TCZ = CZ \tag{99(c)}$$

$$TCL = CL + p\frac{\partial CL}{\partial p} \tag{99(d)}$$

$$TCM = CM + q\frac{\partial CM}{\partial q} \tag{99(e)}$$

and

$$TCN = CN + p\frac{\partial CN}{\partial p} - r\frac{\partial CN}{\partial r}, \tag{99(f)}$$

where CX, CY etc. are read from the data store and the dynamic derivatives are obtained via equations (98) also from the data store. It is assumed that $\frac{\partial CN}{\partial r} \equiv \frac{\partial CM}{\partial q}$. Finally, CREF transforms the six calculated coefficients from total axes to bomb axes and then returns control to the Main Program.

Listings of CORECT and CREF are given in Appendix VII.

4.7 The non-uniform flow routine CINC

Subroutine CINC uses the methods of Section 3.2 to calculate non-uniform flow increments to the coefficients calculated in Section 4.6.

Since Section 3.2 fully discusses the techniques involved, it is sufficient to refer the reader to that section rather than repeat the description here.

The derivatives $\frac{dS}{dx}$, $\frac{dV}{dx}$, $\frac{dW}{dx}$ required by equations (76) and (81) are calculated in the following way:

(i) If the derivative $\frac{dF}{dx}$ is required at the point x_i, then CINC fits the parabola

$$F = a + bx + cx^2$$

successively through the points (x_{i-2}, x_{i-1}, x_i), (x_{i-1}, x_i, x_{i+1}) and (x_i, x_{i+1}, x_{i+2}).

(ii) Each estimate of the derivative is then given by

$$\frac{dF}{dx} = b + 2cx.$$

(iii) The arithmetic mean of the three estimates is then calculated to give the final estimate of $\frac{dF}{dx}$ at the point x_i. This averaging technique gives good results and seems to obviate the problems which arise in a simple curve fit when the points are unevenly spaced.

When CINC has calculated the non-uniform flow force and moment increments at each of the bomb measurement stations (x_i), it integrates each contribution with respect to x to give the non-uniform flow and moment increments for the entire bomb.

A listing of Subroutine CINC appears in Appendix VII.

4.8 The integration routines INTM and DAUX

Subroutine INTM is simply a classical fourth-order Runge-Kutta integrator similar to those which have been in existence for some years now.

Briefly, let the system of equations to be solved be given in the form

$$\dot{x} = f(x,t).$$

If the point t has been reached and x(t) = x, INTM calculates in succession the quantities

$$k_0 = hf(x, t),$$

$$k_1 = hf(x + \tfrac{1}{2}k_0, t + \tfrac{1}{2}h),$$

$$k_2 = hf(x + \tfrac{1}{2}k_1, t + \tfrac{1}{2}h),$$

and

$$k_3 = hf(x + k_2, t + h),$$

where h is the time step.

Then

$$x(t + h) = x(t) + (k_0 + 2k_1 + 2k_2 + k_3)/6.$$

In fact, INTM integrates 18 first-order differential equations at each time step. To do this it uses the first 75 elements of the main program matrix T.

T(1) is used to store time, T(2) is the current time and T(3) is the time step.

T(4) to T(21) are the dependent variables to be integrated, where

(a) T(4) to T(12) are the elements C(1,1), C(2,1), C(3,1), C(1,2), ... C(3,3) of the direction-cosine matrix C as defined in equations (4) and (47).

(b) T(13) to T(15) are X, Y, Z, the position of the bomb centre of gravity relative to aircraft axes,

(c) T(16) to T(18) are U, V, W, the velocity of the bomb centre of gravity relative to aircraft axes,

(d) T(19) to T(21) are the angular momentum components of the bomb relative to earth axes, but expressed in bomb axes,

(e) T(22) to T(30) are the rates of change of the direction-cosines in (a) above, as given by equation (54),

(f) T(31) to T(33) are the rates of change of X, Y, Z in (b) above,

(g) T(34) to T(36) are the rates of change of U, V, W in (c) above, and

(h) T(37) to T(39) are the rates of change of the angular momentum components in (d) above.

In addition, INTM uses T(40) to T(75) as working storage.

Whenever INTM wishes to calculate the derivatives corresponding to the 18 first-order differential equations it calls Subroutine DAUX.

DAUX first updates the angular velocities p, q, r from the angular momentum components by means of equation (40).

It also re-normalizes the elements of the direction-cosine matrix by means of the formula

$$C'_{ij} = C_{ij} \Bigg/ \sqrt{\sum_{j=1}^{3} C^2_{ij}} \qquad (i = 1,2,3).$$

The truth of this relationship follows from equation (4), which shows that the sums of the squares of the elements in any row or column of the direction-cosine matrix C is unity.

Next DAUX updates the bomb and aircraft rotation matrices Ω_B and Ω_A from the angular rates p, q, r and p_A, q_A, r_A respectively, according to equation (15).

Then the following derivatives are calculated:

(i) The direction-cosine matrix derivatives $(\dot{C})$, one for each of the nine elements, given by equation (54) as

$$\dot{C} = \Omega_B C - C\,\Omega_A. \tag{100}$$

(ii) The derivative $(\dot{\underline{X}}_A)$ of (X,Y,Z), given by equation (34) as

$$\dot{\underline{X}}_A = \underline{u} + \Omega_A \underline{X}_A. \tag{101}$$

(iii) The derivative ($\dot{\underline{u}}$) of (U, V, W), given by equation (39) as

$$\dot{\underline{u}} = \underline{F}_A/m - \begin{pmatrix} \mu g \sin(\mathrm{ATTACK}) \\ 0 \\ -\mu g \cos(\mathrm{ATTACK}) \end{pmatrix} + \Omega_A \underline{u} . \tag{102}$$

(iv) The derivatives ($\dot{\underline{h}}_B$) of the angular momentum components, given by equation (46) as

$$\dot{\underline{h}}_B = \underline{M}_B + \Omega_B \underline{h}_B. \tag{103}$$

As noted in Section 4.2, the forces ($\underline{F}_A$) in equation (102) and the moments ($\underline{M}_B$) in equation (103) are calculated by DAUX via a parabolic extrapolation into the current time interval, in order to overcome an inherent instability in the integration process.

It should be noticed that, once the "bomb-alone trajectory" phase has begun, the variables describing the state of the bomb are referred to earth axes, and therefore DAUX then ignores the final terms of equations (100), (101) and (102). That is, the 3 x 3 matrix $\Omega_A = 0$.

A listing of Subroutines INTM and DAUX is given in Appendix VIII.

5. CONCLUDING COMMENTS

The Captive Trajectory Yawmeter System appears to be a successful means of simulating store separation from aircraft. Results to date are most encouraging. The technique of using a yawmeter probe in place of a model mounted upon a sting decreases problems, such as scale effects, associated with small wind tunnels.

Since a typical trajectory requires approximately 30 min of running time, the CTYS is aimed at small continuous flow facilities. Because a settling delay of approximately 5 s is required at each of the 12 measurement stations, probe settling time therefore accounts for one minute per trajectory point. When, in addition, the traversing time of the rig is taken into account, it is obvious that the quoted total time of 30 min cannot be reduced for any reasonable trajectory, despite the fact that data store accesses have been streamlined to the utmost and are carried out in parallel with the traversing and settling operations. Thus, the CTYS would be clearly unsuited to blowdown tunnels and very expensive to use in large continuous flow facilities.

Two extensions of the system are currently underway. Firstly, a model of the ejectors is being developed(ref.4) so that store position, attitude and rates at the end of the ejector stroke can be fed automatically to the program. At present, these must be supplied by the experimenter as input parameters.

Secondly, it has been found(ref.2) that the yawmeter probe can give spurious readings when it is positioned too close to any part of the aircraft surface, such as bomb racks. Consequently, off-line measurements are currently being made, to provide a data store of aerodynamic coefficients as a function of position and attitude whilst the probe is located within this difficult region.

NOTATION

AROLL,APITCH,AYAW	Euler Roll, Pitch and Yaw of aircraft relative to Earth axes
C	direction-cosine (coordinate transformation) matrix
C_{D_C}	cross-flow drag coefficient
C_{ij}	ij^{th} element of C
D	cross-sectional diameter of bomb (metres)
DIVE	aircraft dive angle
$\underline{F}$	external forces on bomb
I_{XX}, I_{YY}, I_{ZZ}	moments of inertia of bomb about its principal axes (kgm m^2)
K	mean chord of aerofoil
M	Mach number
$\underline{M}$	external moments on bomb (newton-m)
S	cross-sectional area of bomb (= π $D^2/4$)
T	ambient temperature
U,V,W	velocity components of bomb C.G. relative to Aircraft axes (m/s)
X,Y,Z	coordinate axes; position of bomb C.G. (metres)
X*,Y*,Z*	position of aircraft C.G. (metres)
XF,YF,ZF	coordinates of bomb's reference fin relative to Aircraft axes (metres)
XN,YN,ZN	coordinates of bomb's nose relative to Aircraft axes
Y,Z	cross-force components (see Section 3.2)
a	speed of sound (m/s)
$\underline{a}$	vector fixed in Aircraft axes
$\underline{b}$	vector fixed in Bomb axes
$\underline{e}$	vector fixed in Earth axes
f	potential cross-force (see equation (75))
g	gravitational acceleration (9.80665 m/s)
h	height of aircraft; mean height of aerofoil

$\underline{h}$	angular momentum vector (newton-m-s)
m	mass of the bomb (kgm)
p	static pressure (newton/m^2)
p,q,r	angular rates of rotating system about its own X,Y,Z axes respectively
q	dynamic pressure ($\frac{1}{2} \rho V^2$) newton/m^2)
t	time (s)
$\underline{u}$	see equation (33)
x,y,z	position of aircraft C.G. relative to earth axes
Ω	angular rotation matrix
α	incidence
α_e	effective incidence ($\alpha + 2\gamma$)
β	sideslip
β_e	effective sideslip ($\beta + 2\gamma$)
γ	wing camber ($\approx$h/c); ratio of specific heats
φ,θ,β	Euler Roll, Pitch and Yaw
$\varphi^*,\theta^*,\beta^*$	Projected Roll, Pitch and Yaw
η	crossflow-drag proportionality factor (see Section 3.2)
μg	centripetal acceleration of aircraft trajectory
ρ	local air density; radius of curvature
$\underline{\omega}$	angular rotation rate vector

Subscripts

A	Aircraft axes
B	Bomb axes; Buoyancy
E	Earth axes
P	Potential; Probe
T	Total axes
V	Viscous
u	Uniform Flow
0	initial conditions
∞	free-stream conditions

REFERENCES

No.	Author	Title
1	Landers, E.R.A.	"Store Separation from Aircraft Using a Captive Trajectory Yawmeter System". Proceedings of TTCP Technical Panel W-2 Meeting, DREV Valcartier, Quebec, 1974
2	Landers, E.R.A., Simpson, G.J. and Bishop, G.	"Store Separation from Aircraft Using a Captive Trajectory Yawmeter System". WRE-TN-1522 (WR&D), October 1975 Also Proceedings of TTCP Technical Panel W-2 Meeting, RAE Farnborough, 1975
3	Landers, E.R.A.	"WRE Store Separation Programme". Proceedings of TTCP Technical Panel W-2 Meeting, WRE, Salisbury, South Australia, 1977
4	Simpson, G.J.	"Ejection Phase of WRE Store Separation Program". Proceedings of TTCP Technical Panel W-2 Meeting, WRE, Salisbury, South Australia, 1977
5	Heron, I.C.	"The Complete Software for the S1 Wind Tunnel Data Acquisition and Control System". WRE Report 1484 (WR&D), September 1975
6	Glauert, H.	"The Elements of Aerofoil and Airscrew Theory". Second Edition, Cambridge University Press, 1948
7	Allen, H.J. and Perkins, W.	"A Study of Effects of Viscosity on Flow over Slender Inclined Bodies of Revolution". NACA Report 1048, 1951
8	Ward, G.N.	"Supersonic Flow Past Slender Pointed Bodies". Quar. Jour. of Mech. and Applied Maths., Vol.2, Part I, March 1949, pp 75 to 97
9	Jorgensen, L.H.	"Prediction of Static Aerodynamic Characteristics for Space-Shuttle-like and other bodies at Angles of Attack from 0° to 180°". NASA Technical Note D-6996, January 1973
10	Goodwin, F.K. Dillenius, M.F.E. and Nielsen, J.N.	"Prediction of Six-Degree-of-Freedom Store Separation Trajectories at Speeds up to the Critical Speed, Vol.1 - Theoretical Methods and Comparison with Experiment". NEAR Report TR 37, October 1972
11	-	"DOS-15 System Manual". DEC-15-ODFFA-B-D, Digital Equipment Corporation, Maynard, Massachusetts

APPENDIX I

LISTINGS OF THE CONSTANTS EDITOR (EDCON)

```
C EDCON                                         DATE: 22-6-77
C
C CTS CONSTANTS EDITOR
C
C       CALLING SEQUENCE:
C               $A RKA -5
C               $GLOAD
C               LOADER V3A000
C               >_EDCON <ALTMODE>
C
      DIMENSION FILE(2)
      DIMENSION XMS(12),DMS(12),AMS(12)
      DIMENSION T(320),IT(5),TS(17)
      DIMENSION C(3,3),PV(3),DELPV(3),PAV(3)
      DIMENSION FSV(3,6),TIMES(3),ABC(3,6)
C
C*****TIME,INTEGRATION TIME STEP
      EQUIVALENCE (T(2),TIME),(T(3),DT)
C*****DIRECTION COSINES DEFINING ORIENTATION OF BOMB WITH RESPECT TO
C     AIRCRAFT.
      EQUIVALENCE (T(4),C11,C),(T(5),C21),(T(6),C31)
      EQUIVALENCE (T(7),C12),(T(8),C22),(T(9),C32)
      EQUIVALENCE (T(10),C13),(T(11),C23),(T(12),C33)
C*****POSITION OF BOMB CENTRE OF GRAVITY IN AIRCRAFT AXES
      EQUIVALENCE (T(13),X),(T(14),Y),(T(15),Z)
C*****VELOCITY OF BOMB C.G. IN AIRCRAFT AXES.
      EQUIVALENCE (T(16),U),(T(17),V),(T(18),W)
C*****ANGULAR MOMENTUM OF BOMB IN BOMB AXES
      EQUIVALENCE (T(19),HX),(T(20),HY),(T(21),HZ)
C*****RATE OF CHANGE OF DIRECTION COSINES ABOVE
      EQUIVALENCE (T(22),DC11),(T(23),DC21),(T(24),DC31)
      EQUIVALENCE (T(25),DC12),(T(26),DC22),(T(27),DC32)
      EQUIVALENCE (T(28),DC13),(T(29),DC23),(T(30),DC33)
C*****RATE OF CHANGE OF X,Y,Z,ABOVE
      EQUIVALENCE (T(31),DX),(T(32),DY),(T(33),DZ)
C*****RATE OF CHANGE OF U,V,W ABOVE
      EQUIVALENCE (T(34),DU),(T(35),DV),(T(36),DW)
C*****RATE OF CHANGE OF HX,HY,HZ ABOVE
      EQUIVALENCE (T(37),DHX),(T(38),DHY),(T(39),DHZ)
C*****T(40) TO T(75) RESERVED FOR INTM.
C*****AIRCRAFT ROTATION ANGLES W.R.T. EARTH AXES
      EQUIVALENCE (T(76),AROLL),(T(77),APITCH),(T(78),AYAW)
C*****BOMB MOMENTS OF INERTIA
      EQUIVALENCE (T(79),AXX),(T(80),AYY),(T(81),AZZ)
C*****BOMB MASS (KGM)
      EQUIVALENCE (T(82),STMASS)
C*****PROBE ROLL AND PITCH ANGLES W.R.T. TRAVERSE AXES
      EQUIVALENCE (T(83),APTROL),(T(84),APTPIT)
C*****TIME LIMIT FOR DROP,AIR TEMPERATURE (DEG C),
C*****BOMB C.G. (METRES FROM TAIL),AIRCRAFT HEIGHT(METRES),
C*****BOMB SCALE (AS IN 1/SCALE),T(90) SPARE,SPEED OF SOUND,
C*****ACCELERATION DUE TO GRAVITY,T(93) SPARE
      EQUIVALENCE (T(85),TLIMIT),(T(86),TEMP),(T(87),XCG),
     1 (T(88),HGHT0),(T(89),SCALE),
     2 (T(91),VSOUND),(T(92),GRAVAC)
C*****TOTAL PITCH AND ROLL OF BOMB IN FLOWFIELD AT REFERENCE POINT.
      EQUIVALENCE (T(94),THETOT),(T(95),PHITOT)
```

```
C*****FREE STREAM MACH NUMBER,MAX BOMB AREA,MAX BOMB DIAMETER
      EQUIVALENCE (T(96),RMACH0),(T(97),AMAX),(T(98),DMAX)
C*****AIRCRAFT ANGLE OF ATTACK
      EQUIVALENCE (T(99),ATTACK)
C
C*****T(100) TO T(103) SPARE.
C
C*****FREE STREAM DYNAMIC PRESSURE
      EQUIVALENCE (T(104),QDP0)
C*****WIND FORCES ON BOMB IN AIRCRAFT AXES.
      EQUIVALENCE (T(105),WX),(T(106),WY),(T(107),WZ)
C*****WIND MOMENTS ON BOMB IN BOMB AXES.
      EQUIVALENCE (T(108),WL),(T(109),WM),(T(110),WN)
C*****AIRCRAFT ROLL,PITCH,YAW RATE ABOUT AIRCRAFT AXES
      EQUIVALENCE (T(111),PA,PAV),(T(112),QA),(T(113),RA)
C
C***** FREE = -1.(+1.) IF BOMB-ALONE TRAJECTORY
C      IS NOT (IS) OPERATING.
C
      EQUIVALENCE (T(114),FREE)
C*****ALPHA AND BETA OF BOMB USED FOR CALCULATING COEFFICIENTS
      EQUIVALENCE (T(115),ALPHA),(T(116),BETA)
C*****AIRCRAFT ACCELERATIONS IN AIRCRAFT AXES.
      EQUIVALENCE (T(117),AX),(T(118),AY),(T(119),AZ)
C*****FORCES ON BOMB C.G. IN AIRCRAFT AXES.
      EQUIVALENCE (T(120),FX),(T(121),FY),(T(122),FZ)
C*****MOMENTS ON BOMB ABOUT BOMB X,Y,Z AXES
      EQUIVALENCE (T(123),FL),(T(124),FM),(T(125),FN)
C*****ROTATION ANGLES DEFINING ORIENTATION OF BOMB W.R.T.AIRCRAFT.
      EQUIVALENCE (T(126),ROLL),(T(127),PITCH),(T(128),YAW)
C*****ROLL,PITCH,YAW RATES OF BOMB ABOUT BOMB X,Y,Z AXES
      EQUIVALENCE (T(129),P,PV),(T(130),Q),(T(131),R)
C*****PRESSURE AT HEIGHT,DYNAMIC PRESSURE AT HEIGHT.
      EQUIVALENCE (T(132),PRESS),(T(133),QDPALT)
C*****DEGREES TO RADIANS,RADIANS TO DEGREES
      EQUIVALENCE (T(134),DTR),(T(135),RTD)
C*****VELOCITY OF AIR AT -INFINITY-
      EQUIVALENCE (T(136),VELINF)
C*****AIRCRAFT SELF-ROTATION-RATE MATRIX.
      EQUIVALENCE (T(137),AR11),(T(138),AR21),(T(139),AR31)
      EQUIVALENCE (T(140),AR12),(T(141),AR22),(T(142),AR32)
      EQUIVALENCE (T(143),AR13),(T(144),AR23),(T(145),AR33)
C*****BOMB SELF-ROTATION-RATE MATRIX.
      EQUIVALENCE (T(146),BR11),(T(147),BR21),(T(148),BR31)
      EQUIVALENCE (T(149),BR12),(T(150),BR22),(T(151),BR32)
      EQUIVALENCE (T(152),BR13),(T(153),BR23),(T(154),BR33)
C*****T(155) TO T(160) SPARE.
C
C*****GRAVITY FORCE IN AIRCRAFT AXES.
      EQUIVALENCE (T(161),GX),(T(162),GY),(T(163),GZ)
C*****DIVE ANGLE OF AIRCRAFT.
      EQUIVALENCE (T(164),DIVE)
C*****PI,TWOPI,CONVERSION FROM METRES TO INCHES
      EQUIVALENCE (T(165),PI),(T(166),TWOPI),(T(167),CMI)
C****  CENTRIPETAL ACCELERATION OF AIRCRAFT (IN G'S)
      EQUIVALENCE (T(168),GF)
C*****AT ALL MEASUREMENT STATIONS...AREAS,DIAMETERS,DYNAMIC PRESSURE,
C*****MACH NUMBER,U,V,W (WIND VELOCITY COMPONENTS,BOMB AXES),
C*****AND MEASUREMENT STATIONS (C.G.=0,NOSE +VE,I.E.,BOMB AXES)
      EQUIVALENCE (T(169),AMS),(T(181),DMS),(T(193),QMS),
```

```
      1 (T(205),RMS),(T(217),UMS),(T(229),VMS),(T(241),WMS),
      2 (T(253),XMS)
C*****FREE STREAM AERODYNAMIC COEFFS BOMB AXES
      EQUIVALENCE (T(265),CX),(T(266),CY),(T(267),CZ),
      1 (T(268),CL),(T(269),CM),(T(270),CN)
C*****INCREMENTS TO COEFFS DUE TO NON-UNIFORM FLOW
      EQUIVALENCE (T(271),DCX),(T(272),DCY),(T(273),DCZ),
      1 (T(274),DCL),(T(275),DCM),(T(276),DCN)
C*****PROBE PRESSURE MEASUREMENTS
      EQUIVALENCE (T(277),DP13),(T(278),DP24),
      1 (T(279),PMAN),(T(280),PPIT),(T(281),TPRESS)
C     AREA TO SAVE COEFFICIENTS AT PREVIOUS 2 TRAJ. POINTS.
      EQUIVALENCE (T(282),FSV),(T(300),TIMES),(T(303),ABC)
C
C
C*****NO OF MEASUREMENT STATIONS,REF POINT,INTEGRATION LIMITS.
      EQUIVALENCE (IT(1),NMS),(IT(2),NREF),(IT(3),NBACK)
      EQUIVALENCE (IT(4),NFRONT),(IT(5),NPRINT)
C
      EQUIVALENCE (TS(4),ROLLS),(TS(5),PITCHS),(TS(6),YAWS)
      EQUIVALENCE (TS(7),PS),(TS(8),QS),(TS(9),RS)
      EQUIVALENCE (TS(10),ATTACS),(TS(11),DIVES)
      EQUIVALENCE (TS(12),PAS),(TS(13),QAS),(TS(14),RAS)
      EQUIVALENCE (TS(15),SPTROL),(TS(16),SPTPIT)
      EQUIVALENCE (DELPV(1),DELP),(DELPV(2),DELQ),(DELPV(3),DELR)
C
C
C CONSTANT DATA:
C
      DATA ANSNO/5HN    /,ANSYES/5HY    /,NMS/12/
      DATA NREF/2/,NBACK/3/,NFRONT/12/
      DATA PI/3.14159265/,CMI/39.370079/,GRAVAC/9.80665/
      DATA DTR/0.017453293/,RTD/57.29578/
C
      TAN(X)=SIN(X)/COS(X)
C
C
1     FORMAT(///1X'CTS CONSTANTS EDITOR'//1X'ENTER FILE NAME   ')
2     FORMAT(A5,A4)
3     FORMAT(//1X,'CREATING NEW FILE?')
4     FORMAT(//1X'***BOMB PARAMETERS***')
714   FORMAT(/1X'MOMENTS OF INERTIA')
5     FORMAT(1X'IX (KGM M2):'F12.5)
6     FORMAT(F12.6)
7     FORMAT(1X'IY (KGM M2):'F12.5)
8     FORMAT(1X'IZ (KGM M2):'F12.5)
9     FORMAT(/1X'MASS (KGM):'F12.5)
10    FORMAT(/1X'C.G. POSN (METRES FROM TAIL):'F12.5)
11    FORMAT(/1X'BOMB SCALE (S AS IN 1/S):'F12.5)
12    FORMAT(/1X'ENTER',I4,' MEASUREMENT STATIONS (M)'//)
13    FORMAT(//1X'ENTER',I4,' DIAMETERS AT MEAS. STATIONS (M)'/)
14    FORMAT(//1X'POSN OF BOMB C.G. (AIRCRAFT AXES)')
15    FORMAT(1X'X (METRES):'F12.5)
16    FORMAT(1X'Y (METRES):'F12.5)
17    FORMAT(1X'Z (METRES):'F12.5)
18    FORMAT(//1X' VELOCITY OF BOMB C.G. (AIRCRAFT AXES)')
19    FORMAT(1X'U (M/S):'F12.5)
20    FORMAT(1X'V (M/S):'F12.5)
21    FORMAT(1X'W (M/S):'F12.5)
121   FORMAT(//1X' BOMB ATTITUDE (AIRCRAFT AXES)')
```

```
22      FORMAT(//1X'ANGULAR RATES (BOMB AXES)')
23      FORMAT(1X'ROLL RATE (DEG/S):'F12.5)
24      FORMAT(1X'PITCH RATE (DEG/S):'F12.5)
25      FORMAT(1X'YAW RATE (DEG/S):'F12.5)
26      FORMAT(//1X'***AIRCRAFT PARAMETERS ****')
126     FORMAT(/1X'HEIGHT (METRES):'F12.5)
27      FORMAT(/1X'ANGLE OF ATTACK (DEG):'F12.5)
28      FORMAT(/1X'DIVE ANGLE (DEG):'F12.5)
29      FORMAT(//1X'ANGULAR RATES (AIRCRAFT AXES):')
30      FORMAT(//1X'***PROBE ATTITUDE (TUNNEL AXES)***')
31      FORMAT(1X'ROLL ANGLE (DEG):'F12.5)
32      FORMAT(1X'PITCH ANGLE (DEG):'F12.5)
33      FORMAT(1X'YAW ANGLE (DEG):'F12.5)
37      FORMAT(//1X'***MISCELLANEOUS INFORMATION***')
38      FORMAT(1X"CENTRIPETAL ACCN OF AIRCRAFT (IN G'S):"F12.5)
39      FORMAT(1X'TIME STEP (S):'F12.5)
40      FORMAT(1X'FREE STREAM MACH NO.:'F12.5)
42      FORMAT(/1X"BOMB-ALONE PRINT INCREMENT (IN DT'S):"I4)
41      FORMAT(1X'TIME LIMIT FOR DROP (S):'F12.5)
C
1001    FORMAT(/1X'ALTER PARAMETERS?')
1002    FORMAT(/1X'ENTER N,WHERE N='/1X'1  BOMB;2  AIRCRAFT;3  PROBE;
        14  MISCELLANEOUS;'
        2 /1X'5  MEASUREMENT STATIONS;6  DIAMETERS AT M.S.'/)
1003    FORMAT(I2)
1004    FORMAT(/1X'ENTER NUMBER OF CHANGES')
1005    FORMAT(/1X'XMS(L):  ENTER L,CARRIAGE RETURN,THEN VALUE')
1006    FORMAT(/1X'DMS(L):  ENTER L,CARRIAGE RETURN,THEN VALUE')
1007    FORMAT(/1X'FILE NAME NOT KNOWN - TRY AGAIN!'//)
1008    FORMAT(/1X'ENTER N,AS DEFINED ABOVE')
1009    FORMAT(1X':')
1010    FORMAT(/1X'FILE ALREADY PRESENT ON DISK - DO YOU WISH
        1 TO DISCARD IT? (Y/N)')
C
101     FORMAT(1X'CTYS CONSTANTS FOR FILE ',A5,A4//)
C
4002    FORMAT(A1)
C
C
        TWOPI=2.*PI
        PI4=PI/4.
        IOLD=1
        IPRTD=1
C
C       ASK FOR FILE NAME
C
1099    WRITE(4,1)
        READ(7,2)FILE
C
C       ASK IF CREATING NEW FILE.
        IRST=1
        CALL CTRLP
        GO TO (201,202,203,204,205,206,207),IRST
201     WRITE(4,3)
        READ(7,4002)ANS
        IF(ANS.EQ.ANSNO) GO TO 1000
        CALL FSTAT(2,FILE,IPRES)
        IF(IPRES.EQ.0) GO TO 302
        WRITE(4,1010)
        READ(7,4002) ANS
```

```
      IF(ANS.EQ.ANSYES) GO TO 302
      CALL CLOSE(2)
      GO TO 1099
C
C     YES. NEW FILE.
302   IRST=2
202   WRITE(4,4)
C
C     ASK FOR POSITION OF BOMB C.G. (AIRCRAFT AXES).
      WRITE(4,14)
      WRITE(4,15) X
      CALL DECDE(X)
      WRITE(4,16) Y
      CALL DECDE(Y)
      WRITE(4,17) Z
      CALL DECDE(Z)
C
C     ASK FOR VELOCITY OF BOMB C.G. (AIRCRAFT AXES).
      WRITE(4,18)
      WRITE(4,19) U
      CALL DECDE(U)
      WRITE(4,20) V
      CALL DECDE(V)
      WRITE(4,21) W
      CALL DECDE(W)
C
C     ASK FOR BOMB ATTITUDE (AIRCRAFT AXES).
      WRITE(4,121)
      WRITE(4,31)ROLLS
      CALL DECDE(ROLLS)
      WRITE(4,32) PITCHS
      CALL DECDE(PITCHS)
      WRITE(4,33) YAWS
      CALL DECDE(YAWS)
C
C     ASK FOR ANGULAR RATES (BOMB AXES).
      WRITE(4,22)
      WRITE(4,23) PS
      CALL DECDE(PS)
      WRITE(4,24) QS
      CALL DECDE(QS)
      WRITE(4,25) RS
      CALL DECDE(RS)
C
C     ASK FOR BOMB PRINCIPAL MOMENTS OF INERTIA.
C
      WRITE(4,714)
C     IX (KGM M2)
      WRITE(4,5) AXX
      CALL DECDE(AXX)
C
C     IY (KGM M2)
      WRITE(4,7) AYY
      CALL DECDE(AYY)
C
C     IZ (KGM M2)
      WRITE(4,8) AZZ
      CALL DECDE(AZZ)
C
C     MASS OF BOMB (KGM)
```

```
      WRITE(4,9) STMASS
      CALL DECDE(STMASS)
C
C     C.G. POSITION (METRES FROM TAIL)
      WRITE(4,10) XCG
      CALL DECDE(XCG)
C
C     SCALE,AS IN 1/SCALE.
      WRITE(4,11) SCALE
      CALL DECDE(SCALE)
      GO TO (303,73),IOLD
C
C     GET THE 'NMS' MEASUREMENT STATIONS.
303   IRST=3
203   WRITE(4,12)NMS
      DO 500 L=1,NMS
500   CALL DECDE(XMS(L))
C
C     GET DIAMETERS AT THESE MEASURED STATIONS.
      IRST=4
204   WRITE(4,13)NMS
      DO 501 L=1,NMS
501   CALL DECDE(DMS(L))
C
C*******ACQUIRE AIRCRAFT PARAMETERS******
C
305   IRST=5
205   WRITE(4,26)
C
C     AIRCRAFT HEIGHT (METRES)
      WRITE(4,126) HGHT0
      CALL DECDE(HGHT0)
C
C     ANGLE OF ATTACK (DEGREES).
      WRITE(4,27) ATTACS
      CALL DECDE(ATTACS)
      ATTACK=ATTACS*DTR
C
C     DIVE ANGLE (DEGREES).
      WRITE(4,28) DIVES
      CALL DECDE(DIVES)
      DIVE=DIVES*DTR
C
      GO TO (306,73),IOLD
C
C*******ASK FOR PROBE ATTITUDE******
C
306   IRST=6
206   WRITE(4,30)
C
C     PROBE ROLL (DEGREES)
      WRITE(4,31) SPTROL
      CALL DECDE(SPTROL)
      APTROL=SPTROL*DTR
C
C     PROBE PITCH (DEGREES).
      WRITE(4,32) SPTPIT
      CALL DECDE(SPTPIT)
      APTPIT=SPTPIT*DTR
      GO TO (307,73),IOLD
```

```
C
C
C*****ACQUIRE MISCELLANEOUS INFORMATION*****
C
307      IRST=7
207      WRITE(4,37)
C
C        CENTRIPETAL ACCN. OF AIRCRAFT (IN G'S).
         WRITE(4,38) GF
         CALL DECDE(GF)
C
C        TIME STEP (SECONDS)
         WRITE(4,39) DT
         CALL DECDE(DT)
C
C        FREE STREAM MACH NUMBER.
         WRITE(4,40) RMACH0
         CALL DECDE(RMACH0)
C
C        NUMBER OF TIME INCREMENTS BEFORE PRINTING
C        (DURING BOMB-ALONE TRAJECTORY).
C
         WRITE(4,42) NPRINT
         CALL IDECD(NPRINT)
C
C        TIME LIMIT FOR DROP (SECONDS).
         WRITE(4,41) TLIMIT
         CALL DECDE(TLIMIT)
         GO TO (6666,73),IOLD
C
C*********
C*********
C*********MODIFY EXISTING CONSTANTS BLOCK****
C
C
1000     IOLD=2
         CALL FSTAT(2,FILE,IPRES)
         IF(IPRES.NE.0) GO TO 72
         WRITE(4,1007)
         GO TO 1099
72       CALL SEEK(2,FILE)
         READ(2)T,IT,TS
         CALL CLOSE(2)
         CALL CTRLP
73       WRITE(4,1001)
         READ(7,4002) ANS
         IF(ANS.EQ.ANSNO) GO TO 6666
         GO TO (1100,1101),IPRTD
1100     IPRTD=2
         WRITE(4,1002)
1101     WRITE(4,1008)
         READ(7,1003) NSYS
         IF(NSYS.LT.1.OR.NSYS.GT.6) GO TO 1100
         GO TO (302,305,306,307,309,310),NSYS
309      WRITE(4,1004)
         READ(7,1003) NCH
         IF(NCH.LE.0.OR.NCH.GT.NMS) GO TO 73
         WRITE(4,1005)
         DO74 JJ=1,NCH
         WRITE(4,1009)
         READ(7,1003) L
74       CALL DECDE(XMS(L))
         GO TO 73
```

```
C
310     WRITE(4,1004)
        READ(7,1003) NCH
        IF(NCH.LE.0.OR.NCH.GT.NMS) GO TO 73
        WRITE(4,1006)
        DO75 JJ=1,NCH
        WRITE(4,1009)
        READ(7,1003) L
75      CALL DECDE(DMS(L))
        GO TO 73
C
C
C FIND EULER ROLL, PITCH & YAW.
C
C
6666    ROLL=ROLLS*DTR
        PITCHR=PITCHS*DTR
        TPCH=TAN(PITCHR)
        YAWSR=YAWS*DTR
        TYW=TAN(YAWSR)
        SRL=SIN(ROLL)
        CRL=COS(ROLL)
        TANTH=TPCH*CRL+TYW*SRL
        PITCH=ATAN(TANTH)
        COSTH=COS(PITCH)
        TANB=(TYW*CRL-TPCH*SRL)*COSTH
        YAW=ATAN(TANB)
C
C CALCULATE FREE STREAM VELOCITY
C
        TEMP=14.99-0.0065*HGHT0
        VSOUND=SQRT(401.742*TEMP+109809.48)
        VELINF=VSOUND*RMACH0
C
C CALCULATE AIRCRAFT ANGULAR RATES
C       ASSUMES AROLL=PI AND AYAW=0.
C
        PA=0.
        QA=GF*GRAVAC/VELINF
        RA=0.
C
C CONVERT TO DEG/S FOR PRINTING.
C
        PAS=0.
        QAS=QA*RTD
        RAS=0.
C
C       FIND AREAS,MAX AREA,MAX DIAMETER
C
        DMAX=0.
        AMAX=0.
        DO700 L=1,NMS
        AMS(L)=PI4*DMS(L)**2
        IF(DMS(L).LE.DMAX) GO TO 700
        DMAX=DMS(L)
        AMAX=AMS(L)
700     CONTINUE
C
C
C       INITIALIZE DIRECTION COSINES OF BOMB W.R.T. AIRCRAFT.
```

```
C
      SR=SIN(ROLL)
      CR=COS(ROLL)
      SP=SIN(PITCH)
      CP=COS(PITCH)
      SY=SIN(YAW)
      CY=COS(YAW)
      C11=CP*CY
      C12=SR*SP*CY+CR*SY
      C13=-CR*SP*CY+SR*SY
      C21=-CP*SY
      C22=-SR*SP*SY+CR*CY
      C23=CR*SP*SY+SR*CY
      C31=SP
      C32=-SR*CP
      C33=CR*CP
C
C CALCULATE ANGULAR RATES OF BOMB (IN BOMB AXES) W.R.T FIXED
C EARTH AXIS SYSTEM.
C
C     FIRST CALCULATE DELP, DELQ & DELR FROM PROJECTED
C     ROLL, PITCH AND YAW RATES.
C
      RR=PS*DTR
      PP=QS*DTR/COS(PITCHR)**2
      YY=RS*DTR/COS(YAWSR)**2
C
      DELP=C21*(C32*YY-C33*PP)+RR*CP/CY
      DELQ=C11*(C33*PP-C32*YY)
      DELR=C11*(C22*YY-C23*PP)
C
C     NOW ADD (C).(PA), THE AIRCRAFT'S ANGULAR
C     RATES W.R.T. FIXED AXES, TRANSFORMED TO BOMB AXES.
C     (P) = (C)(PA) + (DELP)
C
      DO801 I=1,3
      SUM=0.
      DO800 J=1,3
800   SUM=SUM+C(I,J)*PAV(J)
801   PV(I)=SUM+DELPV(I)
C
C
C     INITIAL ANGULAR MOMENTUM OF BOMB  (KGM M2/S)
C
      HX=AXX*P
      HY=AYY*Q
      HZ=AZZ*R
C
C     CALCULATE AIRCRAFT ACCELERATIONS (IN AIRCRAFT AXES).
C
      GFG=GF*GRAVAC
      AX=GFG*SIN(ATTACK)
      AY=0.
      AZ=-GFG*COS(ATTACK)
C
C     WRITE T BLOCK ON DISK.
C
      CALL ENTER (2,FILE)
      WRITE(2) T,IT,TS
      CALL CLOSE(2)
```

```
C
C
C         PRINT ALL CONSTANTS ON THE LINE PRINTER
C
C         WRITE(9,6749)T
C6749     FORMAT(1X10G13.5)
C
          WRITE(9,101) FILE
          WRITE(9,4)
          WRITE(9,5) AXX
          WRITE(9,7) AYY
          WRITE(9,8) AZZ
          WRITE(9,9) STMASS
          WRITE(9,10) XCG
          WRITE(9,11) SCALE
          WRITE(9,14)
          WRITE(9,15) X
          WRITE(9,16) Y
          WRITE(9,17) Z
          WRITE(9,18)
          WRITE(9,19) U
          WRITE(9,20) V
          WRITE(9,21) W
          WRITE(9,121)
          WRITE(9,31) ROLLS
          WRITE(9,32) PITCHS
          WRITE(9,33) YAWS
          WRITE(9,22)
          WRITE(9,23) PS
          WRITE(9,24) QS
          WRITE(9,25) RS
          WRITE(9,1025) XMS
1025      FORMAT(/1X'MEASUREMENT STATIONS ARE:'/1X12F10.4//)
          WRITE(9,1026) DMS
1026      FORMAT(1X'DIAMETERS AT THESE STATIONS ARE:'/1X12F10.4//)
          WRITE(9,26)
          WRITE(9,126) HGHT0
          WRITE(9,27) ATTACS
          WRITE(9,28) DIVES
          WRITE(9,29)
          WRITE(9,23) PAS
          WRITE(9,24) QAS
          WRITE(9,25) RAS
          WRITE(9,30)
          WRITE(9,31) SPTROL
          WRITE(9,32) SPTPIT
          WRITE(9,37)
          WRITE(9,38) GF
          WRITE(9,39) DT
          WRITE(9,40)RMACH0
          WRITE(9,42) NPRINT
          WRITE(9,41) TLIMIT
          STOP
          END
```

```
C DECDE                                DATE: 12-7-77
C
C ROUTINES TO ACQUIRE A FLOATING POINT OR FIXED
C POINT NUMBER FROM THE TELETYPE.
C       IF THE OPERATOR HAS TYPED A CARRIAGE RETURN ONLY,
C THE VALUE OF THE ARGUMENT IS NOT CHANGED AND AN IMMEDIATE
C RETURN TO THE CALLING PROGRAM IS EFFECTED.
C
C CALLED BY EDCON.
C
C
        SUBROUTINE DECDE (A)
        DIMENSION ASC(3)
        DATA BLANKS/5H     /
C
        I=1
        GO TO 100
C
        ENTRY IDECD (N)
        I=2
100     READ(7,1) ASC
        IF(ASC(1).EQ.BLANKS) GO TO 300
        GO TO (101,102), I
101     DECODE(12,ASC,) A
        GO TO 300
102     DECODE(12,ASC,) N
1       FORMAT(3A5)
300     RETURN
        END
```

APPENDIX II

LISTING OF THE MAIN PROGRAM

```
C MAIN                              DATE: 8-7-77
C
C*****PROGRAM MAIN (WIND TUNNEL VERSION)
C*****ROUTINES REQUIRED...
C        TRANS,PARAB,ARCTAN,CPBIN,READIN,TRFREE,TRSET
C        FLOW,CALC,TRVCTS,PROBE
C        CREF,CINC,CORECT
C        INT,DAUX
C
C*****LIST OF STOP LOCATIONS...
C*****1-PROBE,2-READIN,3-,4-,5-PROBE,6-READIN
C*****7-READIN,8-,9-,10-CREF,11-PROBE.
C
C
         DIMENSION FILE(2),TFILE(2)
         DIMENSION TMACHF(2),PRATFL(2),FILEMC(2),STORCX(2)
         DIMENSION AMK82(2),AKAR(2)
         DIMENSION FVEC(6),FSV(3,6),TIMES(3),ABC(3,6)
         DIMENSION XF(3),XN(3),DUM(3),ELA(3)
C
C
C*****STORAGE FOR AREAS,DIAMETERS,DYNAMIC PRESSURE,MACH NUMBER,
C*****VELOCITY U,V,W, MEASUREMENT STATIONS (AT EACH M.S.,BOMB AXES)
         DIMENSION AMS(12),DMS(12),QMS(12),RMS(12),
         1 UMS(12),VMS(12),WMS(12),XMS(12)
C
C
         COMMON T(320),IT(5),NF
C
C
C*****TIME,INTEGRATION TIME STEP
         EQUIVALENCE (T(2),TIME),(T(3),DT)                         [AIRCRAFT
C*****DIRECTION COSINES DEFINING ORIENTATION OF BOMB WITH RESPECT TO
         EQUIVALENCE (T(4),C11),(T(5),C21),(T(6),C31)
         EQUIVALENCE (T( 7),C12),(T( 8),C22),(T( 9),C32)
         EQUIVALENCE (T(10),C13),(T(11),C23),(T(12),C33)
C*****POSITION OF BOMB CENTRE OF GRAVITY IN AIRCRAFT AXES
         EQUIVALENCE (T(13),X),(T(14),Y),(T(15),Z)
C*****VELOCITY OF BOMB C.G. IN AIRCRAFT AXES.
         EQUIVALENCE (T(16),U),(T(17),V),(T(18),W)
C*****ANGULAR MOMENTUM OF BOMB IN BOMB AXES
         EQUIVALENCE (T(19),HX),(T(20),HY),(T(21),HZ)
C*****RATE OF CHANGE OF DIRECTION COSINES ABOVE
         EQUIVALENCE (T(22),DC11),(T(23),DC21),(T(24),DC31)
         EQUIVALENCE (T(25),DC12),(T(26),DC22),(T(27),DC32)
         EQUIVALENCE (T(28),DC13),(T(29),DC23),(T(30),DC33)
C*****RATE OF CHANGE OF X,Y,Z,ABOVE
         EQUIVALENCE (T(31),DX),(T(32),DY),(T(33),DZ)
C*****RATE OF CHANGE OF U,V,W ABOVE
         EQUIVALENCE (T(34),DU),(T(35),DV),(T(36),DW)
C*****RATE OF CHANGE OF HX,HY,HZ ABOVE
         EQUIVALENCE (T(37),DHX),(T(38),DHY).(T(39),DHZ)
C*****T(40) TO T(75) RESERVED FOR INTM.
C*****AIRCRAFT ROTATION ANGLES W.R.T. EARTH AXES
         EQUIVALENCE (T(76),AROLL),(T(77),APITCH),(T(78),AYAW)
C*****BOMB MOMENTS OF INERTIA
         EQUIVALENCE (T(79),AXX),(T(80),AYY),(T(81),AZZ)
```

```
C*****BOMB MASS (KGM SLUGS)
      EQUIVALENCE (T(82),STMASS)
C*****PROBE ROLL AND PITCH ANGLES W.R.T. TRAVERSE AXES
      EQUIVALENCE (T(83),APTROL),(T(84),APTPIT)
C*****TIME LIMIT FOR DROP,AIR TEMPERATURE (DEG C),
C*****BOMB C.G. (METRES FROM TAIL),AIRCRAFT HEIGHT(METRES),
C*****BOMB SCALE (AS IN 1/SCALE),T(90) SPARE,SPEED OF SOUND,
C*****ACCELERATION DUE TO GRAVITY,T(93) SPARE
      EQUIVALENCE (T(85),TLIMIT),(T(86),TEMP),(T(87),XCG),
     1 (T(88),HGHT0),(T(89),SCALE),
     2 (T(91),VSOUND),(T(92),GRAVAC)
C*****TOTAL PITCH AND ROLL OF BOMB IN FLOWFIELD AT REFERENCE POINT.
      EQUIVALENCE (T(94),THETOT),(T(95),PHITOT)
C*****FREE STREAM MACH NUMBER,MAX BOMB AREA,MAX BOMB DIAMETER
      EQUIVALENCE (T(96),RMACH0),(T(97),AMAX),(T(98),DMAX)
C*****AIRCRAFT ANGLE OF ATTACK
      EQUIVALENCE (T(99),ATTACK)
C*****T(100) TO T(103) SPARE.
C
C*****FREE STREAM DYNAMIC PRESSURE
      EQUIVALENCE (T(104),QDP0)
C*****WIND FORCES ON BOMB IN AIRCRAFT AXES.
      EQUIVALENCE (T(105),WX),(T(106),WY),(T(107),WZ)
C*****WIND MOMENTS ON BOMB IN BOMB AXES.
      EQUIVALENCE (T(108),WL),(T(109),WM),(T(110),WN)
C*****AIRCRAFT ROLL,PITCH,YAW RATE ABOUT AIRCRAFT AXES
      EQUIVALENCE (T(111),PA),(T(112),QA),(T(113),RA)
C***** FREE = -1. (+1.) IF BOMB-ALONE TRAJECTORY
C      IS NOT (IS) OPERATING.
      EQUIVALENCE (T(114),FREE)
C*****ALPHA AND BETA OF BOMB USED FOR CALCULATING COEFFICIENTS
      EQUIVALENCE (T(115),ALPHA),(T(116),BETA)
C*****AIRCRAFT ACCELERATIONS IN AIRCRAFT AXES.
      EQUIVALENCE (T(117),AX),(T(118),AY),(T(119),AZ)
C*****FORCES ON BOMB C.G. IN AIRCRAFT AXES.
      EQUIVALENCE (T(120),FX,FVEC),(T(121),FY),(T(122),FZ)
C*****MOMENTS ON BOMB ABOUT BOMB X,Y,Z AXES
      EQUIVALENCE (T(123),FL),(T(124),FM),(T(125),FN)
C*****ROTATION ANGLES DEFINING ORIENTATION OF BOMB W.R.T.AIRCRAFT.
      EQUIVALENCE (T(126),ROLL),(T(127),PITCH),(T(128),YAW)
C*****ROLL,PITCH,YAW RATES OF BOMB ABOUT BOMB X,Y,Z AXES
      EQUIVALENCE (T(129),P),(T(130),Q),(T(131),R)
C*****PRESSURE AT HEIGHT,DYNAMIC PRESSURE AT HEIGHT.
      EQUIVALENCE (T(132),PRESS),(T(133),QDPALT)
C*****DEGREES TO RADIANS,RADIANS TO DEGREES
      EQUIVALENCE (T(134),DTR),(T(135),RTD)
C*****VELOCITY OF AIR AT -INFINITY-
      EQUIVALENCE (T(136),VELINF)
C*****AIRCRAFT SELF-ROTATION-RATE MATRIX.
      EQUIVALENCE (T(137),AR11),(T(138),AR21),(T(139),AR31)
      EQUIVALENCE (T(140),AR12),(T(141),AR22),(T(142),AR32)
      EQUIVALENCE (T(143),AR13),(T(144),AR23),(T(145),AR33)
C*****BOMB SELF-ROTATION-RATE MATRIX.
      EQUIVALENCE (T(146),BR11),(T(147),BR21),(T(148),BR31)
      EQUIVALENCE (T(149),BR12),(T(150),BR22),(T(151),BR32)
      EQUIVALENCE (T(152),BR13),(T(153),BR23),(T(154),BR33)
C*****T(155) TO T(160) SPARE.
```

```
C*****GRAVITY FORCE IN AIRCRAFT AXES.
      EQUIVALENCE (T(161),GX),(T(162),GY),(T(163),GZ)
C*****ORIGINAL DIVE ANGLE OF AIRCRAFT.
      EQUIVALENCE (T(164),DIVE0)
C*****PI,TWOPI,CONVERSION FROM METRES TO INCHES
      EQUIVALENCE (T(165),PI),(T(166),TWOPI),(T(167),CMI)
C     CENTRIPETAL ACCELERATION OF AIRCRAFT (IN G'S).
      EQUIVALENCE (T(168),GF)
C*****AT ALL MEASUREMENT STATIONS...AREAS,DIAMETERS,DYNAMIC PRESSURE,
C*****MACH NUMBER,U,V,W (WIND VELOCITY COMPONENTS,BOMB AXES),
C*****AND MEASUREMENT STATIONS (C.G.=0,NOSE +VE,I.E.,BOMB AXES)
      EQUIVALENCE (T(169),AMS),(T(181),DMS),(T(193),QMS),
     1 (T(205),RMS),(T(217),UMS),(T(229),VMS),(T(241),WMS),
     2 (T(253),XMS)
C*****FREE STREAM AERODYNAMIC COEFFS BOMB AXES
      EQUIVALENCE (T(265),CX),(T(266),CY),(T(267),CZ),
     1 (T(268),CL),(T(269),CM),(T(270),CN)
C*****INCREMENTS TO COEFFS DUE TO NON-UNIFORM FLOW
      EQUIVALENCE (T(271),DCX),(T(272),DCY),(T(273),DCZ),
     1 (T(274),DCL),(T(275),DCM),(T(276),DCN)
C*****PROBE PRESSURE MEASUREMENTS
      EQUIVALENCE (T(277),DP13),(T(278),DP24),
     1 (T(279),PMAN),(T(280),PPIT),(T(281),TPRESS)
C     AREA TO SAVE COEFFICIENTS AT PREVIOUS 2 TRAJ. POINTS
      EQUIVALENCE (T(282),FSV),(T(300),TIMES),(T(303),ABC)
C
C
C*****NO OF MEASUREMENT STATIONS,REF POINT,INTEGRATION LIMITS.
      EQUIVALENCE (IT(1),NMS),(IT(2),NREF),(IT(3),NBACK)
      EQUIVALENCE (IT(4),NFRONT),(IT(5),NPRINT)
C
      EQUIVALENCE (XF(2),YF),(XF(3),ZF)
      EQUIVALENCE (XN(1),XN1),(XN(2),YN),(XN(3),ZN)
C
      DATA TFILE/5HTFILE,4H2CTS/,ANSYES/5HY    /
C
C*****LABELS FOR YAWMETER PROBE FILES
C
      DATA TMACHF/5HTMACH,4HTPRB/
      DATA PRATFL/5HPRATI,4HOPRB/
      DATA FILEMC/5HMACHC,4HXPRB/
C
C*****LABELS FOR STORE COEFFICIENT FILES
C
      DATA AMK82/5HMK82C,4HXSTC/
      DATA AKAR/5HKARCX,4HSSTC/
C
      DATA ANSM/5HM    /,ANSK/5HK    /
C
      TAN(X)=SIN(X)/COS(X)
C
      IMPACT=1
C
C     SET UP FOR INTERRUPTS FROM TRAVERSE RIG
C     READ BUTTON
C
      CALL TRSET (IFREE)
```

```
C
C
C        SET UP THE PROBE FILES FOR DIRECT ACCESS I/O
C
         CALL DEFINE(14,8,1950,TMACHF,IV1,0,0,0)
C        WAS 2015=65X31; NOW 65X30 (PITCH=30 ABSENT)
C
         CALL DEFINE(3,8,330,PRATFL,IV2,0,0,0)
C        WAS 341=31X11; NOW 30X11 (PITCH=30 ABSENT)
C
         CALL DEFINE(12,10,18491,FILEMC,IV3,0,0,0)
C
C
         WRITE(4,6783)
6783     FORMAT(///1X'CAPTIVE TRAJECTORY YAWMETER SYSTEM'/)
         WRITE(4,6784)
6784     FORMAT(/1X'NEW TRAJ? (Y OR N)')
         READ(7,6795) ANS
6795     FORMAT(A1)
         IF(ANS.EQ.ANSYES) GO TO 8888
         CALL SEEK(2,TFILE)
         READ(2)T,IT,STORCX,INDST,NF
         CALL CLOSE(2)
         GO TO 8496
8888     WRITE(4,6601)
6601     FORMAT(/1X'MK82 (M) OR KARINGA (K)?')
         READ(7,6795) ANS
         IF (ANS.NE.ANSM) GO TO 6602
         STORCX(1)=AMK82(1)
         STORCX(2)=AMK82(2)
         GO TO 8889
6602     IF (ANS.NE.ANSK) GO TO 8888
         STORCX(1)=AKAR(1)
         STORCX(2)=AKAR(2)
8889     WRITE(4,6796)
6796     FORMAT(/1X'ENTER FILE NAME')
6782     READ(7,6794) FILE
6794     FORMAT(A5,A4)
         CALL FSTAT(2,FILE,IPRES)
         IF(IPRES.NE.0) GO TO 6786
         WRITE(4,6785)
6785     FORMAT(/1X'FILE NOT PRESENT - TRY AGAIN!')
         GO TO 6782
6786     CALL SEEK(2,FILE)
         READ(2)T,IT
         CALL CLOSE(2)
         INDST=1
         FREE=-1.
C
8496     CALL DAUX
         GFG=GF*GRAVAC
C*****START THE FLIGHT*
C
3        DIVE=DIVE0-GFG*TIME/VELINF
C
C        AIRCRAFT ORIENTATION W.R.T. EARTH AXES.
```

```
C
      AROLL=PI
      APITCH=-(DIVE-ATTACK)
C
C NOTE: APITCH IS AN EULER ANGLE, WHERE WE HAVE
C       ROLLED THROUGH PI.  HENCE, THE ABOVE
C       FORMULA IS CORRECT EVEN THOUGH IT LOOKS
C       AS THOUGH IT SHOULD HAVE A MINUS THROUGH IT!
C
      AYAW=0.
C
      IF (FREE.GT.0.) GO TO 4
      IF (ABS(GF).GT.1.E-6) GO TO 941
      SV=VELINF*TIME
      VGT=0.
      GO TO 942
941   VG=2.*VELINF/GFG
      VGT=TIME/VG
      SV=VG*VELINF*SIN(VGT)
942   HEIGHT=HGHT0-SV*SIN(DIVE0-VGT)
      IF (IFREE.EQ.0) GO TO 5
      CALL TRFREE (STORCX,HEIGHT)
      INDST=1
      IPRINT=NPRINT-1
4     HEIGHT=Z
C*****CALCULATE QUANTITIES DEPENDING ON HEIGHT OF AIRCRAFT (METRES).
C*****STATIC PRESSURE (NEWTONS/SQ M)
5     PRESS=10335.11*GRAVAC*(1.-0.0000022559*HEIGHT)**5.256103
C*****TEMPERATURE (DEG C)
      TEMP=14.99-0.0065*HEIGHT
C*****VELOCITY OF SOUND (METRES/SEC)
      VSOUND=SQRT(401.742*TEMP+109809.48)
C
      IF (FREE.GT.0.) RMACH0=SQRT(U**2+V**2+W**2)/VSOUND
C*****VELOCITY AT INFINITY (M/S)
      VELINF=VSOUND*RMACH0
C*****DYNAMIC PRESSURE (NEWTONS/SQ M)
      QDPALT=0.5*1.4*PRESS*RMACH0**2
      QAALT=QDPALT*AMAX
      QAD=QAALT*DMAX
C*****FIND AIRCRAFT QUANTITIES IN AIRCRAFT AXES.
C
C
      DUM(1)=0.
      DUM(2)=0.
C
      IF (FREE.GT.0.) GO TO 10
C
C*****FIND GRAVITY IN AIRCRAFT AXES.
C*****GRAVITY FORCE ON BOMB IN EARTH AXES =(0,0,GZE)
      GZE=-STMASS*GRAVAC
      DUM(3)=GZE
C*****FIND GRAVITY FORCE ON BOMB IN AIRCRAFT AXES FROM EARTH AXES
      CALL TRANS (GX,AROLL,APITCH,AYAW,+1,DUM)
      GO TO 11
```

```
C
C*****FIND WIND FORCES IN AIRCRAFT,MOMENTS IN BOMB AXES.
C
10       IPRINT=IPRINT+1
C
C CONVERT VELOCITIES FROM EARTH AXES TO BOMB AXES.
C        BOMB-ALONE TRAJECTORY
         CALL CTRANS (DUM,+1,U)
         UMS(NREF)=-DUM(1)
         VMS(NREF)=-DUM(2)
         WMS(NREF)=-DUM(3)
         VELREF=VELINF
         RMSREF=RMACH0
         RMS(NREF)=RMACH0
         VBETA=VMS(NREF)/UMS(NREF)
         WALPHA=WMS(NREF)/UMS(NREF)
         THETOT=ATAN(SQRT(VBETA**2+WALPHA**2))
         PHITOT=ARCTAN(VBETA,WALPHA)
         RATIO=1.
         CALL CREF
         WXB=CX*QAALT
         WYB=CY*QAALT
         WZB=CZ*QAALT
         WL=CL*QAD
         WM=CM*QAD
         WN=CN*QAD
         GO TO 12
C
C*****FIRST FIND FLOW PROPERTIES ALONG BOMB IN BOMB AXES
11       CALL FLOW
C
C        CALCULATE EFFECTIVE INCIDENCE (ALPHA) AND
C        EFFECTIVE SIDESLIP (BETA) FROM FIN CAMBER
C        AND REFERENCE POINT INCIDENCE AND SIDESLIP.
C
         CALL CORECT(BETA,VMS)
         CALL CORECT(ALPHA,WMS)
C
C*****FIND TOTAL PITCH AND TOTAL ROLL OF BOMB W.R.T. TOTAL AXES AT
C        REFERENCE POINT.
C*****(TOTAL ROLL (PHITOT) IS ANGLE THAT BOMB +Z AXIS
C*****IS ROTATED FROM TOTAL AXIS +Z)
         VBETA=TAN(BETA)
         WALPHA=TAN(ALPHA)
C
C*****FIND TOTAL PITCH ANGLE
         THETOT=ATAN(SQRT(VBETA**2+WALPHA**2))
C*****FIND ROLL ANGLE
         PHITOT=ARCTAN(VBETA,WALPHA)
C*****FIND FORCE AND MOMENT COEFFS DUE TO UNIFORM FLOW EQUAL
C*****TO FLOW AT REFERENCE POINT IN BOMB AXES
         VELREF=SQRT(UMS(NREF)**2+VMS(NREF)**2+WMS(NREF)**2)
         RMSREF=RMS(NREF)
         CALL DEFINE(13,29,1664,STORCX,IV4,0,0,0)
         CALL CREF
         CALL CLOSE(13)
C*****ADJUST TABULATED COEFFS TO CALCULATED COEFFS BY BASING ON
C*****SAME DYNAMIC PRESSURE
         RATIO=QMS(NREF)/QDP0
```

```
C
C*****FIND INCREMENTS TO ABOVE DUE TO FLOW VARIATIONS ALONG BOMB
        CALL CINC
C*****FIND FORCES AND MOMENTS FROM COEFFS AND Q AT ALTITUDE OF FLIGHT
        QAD=QAALT*DMAX
        WXB=CX*RATIO*QAALT
        WYB=(CY*RATIO+DCY)*QAALT
        WZB=(CZ*RATIO+DCZ)*QAALT
        WL=CL*RATIO*QAD
        WM=(CM*RATIO+DCM)*QAD
        WN=(CN*RATIO+DCN)*QAD
C*****EXPRESS FORCES IN AIRCRAFT AXES FROM BOMB AXES
12      CALL CTRANS (WX,-1,WXB)
C*****UPDATE BOMB ORIENTATION IN PROJECTED YAW,PITCH PLANE OF AIRCRAFT
        ROLL=ARCTAN (-C32,C33)
        PITCH=ATAN(-C13/C11)
        YAW=ATAN(C12/C11)
C
C FIND PROJECTED ROLL, PITCH & YAW RATES.
C
        ROLLD=(DC33*C32-DC32*C33)/(C32**2+C33**2)
        PITCHD=(DC11*C13-DC13*C11)/(C11**2+C13**2)
        YAWD=  (DC12*C11-DC11*C12)/(C11**2+C12**2)
C
C*****ADD THE FORCES AND MOMENTS TOGETHER*
        FX=WX+GX
        FY=WY+GY
        FZ=WZ+GZ
        FL=WL
        FM=WM
        FN=WN
C
        TIMES(3)=TIMES(2)
        TIMES(2)=TIMES(1)
        TIMES(1)=TIME
        DO68 L=1,6
C
C       SAVE COEFFICIENTS AT PREVIOUS 2 POINTS.
C
        FSV(3,L)=FSV(2,L)
        FSV(2,L)=FSV(1,L)
        FSV(1,L)=FVEC(L)
        IF(INDST.GT.2) GO TO 67
        ABC(1,L)=FVEC(L)
        ABC(2,L)=0.
        ABC(3,L)=0.
        GO TO 68
67      CALL PARAB (ABC(1,L),TIMES,FSV(1,L))
68      CONTINUE
        INDST=INDST+1
        CALL DAUX
        ROLDEG=ROLL*RTD
        PITDEG=PITCH*RTD
        YAWDEG=YAW*RTD
        PDEG=ROLLD*RTD
        QDEG=PITCHD*RTD
        RDEG=YAWD*RTD
```

```
C
        IF (FREE.GT.0..AND.IPRINT.NE.NPRINT) GO TO 5431
        IPRINT=0
C
5430    IF (FREE.GT.0.) GO TO 69
C
C CONVERT U,V,W INTO PHYSICAL VELOCITIES OF BOMB
C C.G. RELATIVE TO AIRCRAFT AXES, BY ADDING (AROT).(X)
C
        UPR=U+RA*Y-QA*Z
        VPR=V+PA*Z-RA*X
        WPR=W+QA*X-PA*Y
        GO TO 5432
C
69      UPR=U
        VPR=V
        WPR=W
C
5432    WRITE(9,6901) TIME,X,Y,Z,ROLDEG,PITDEG,YAWDEG,
        1 UPR,VPR,WPR,PDEG,QDEG,RDEG
        WRITE(9,6973)FX,FY,FZ,FL,FM,FN
        THETOD=THETOT*RTD
        PHITOD=PHITOT*RTD
        WRITE(9,6873)THETOD,PHITOD,VELREF,RMSREF,RATIO
C
        GO TO (5431,6907), IMPACT
C
6873    FORMAT(5X'THETOT,PHITOT,VELREF,RMSREF,RATIO:    ',2G15.5,3G15.6
        1 ///)
6973    FORMAT(5X'FX,FY,FZ,FL,FM,FN:    ',6G15.6)
6901    FORMAT(1X'TIME,X,Y,Z',F8.4,3X,3G14.5,10X'ROLL,PITCH,YAW',
        1 3G14.5/6X'U,V,W',11X,3G14.5,4X'RATES:ROLL,PITCH,YAW',
        2 3G14.5/)
C*****INTEGRATE ONE TIME STEP*
5431    CALL INTM
        CALL ENTER(2,TFILE)
        WRITE(2) T,IT,STORCX,INDST,NF
        CALL CLOSE(2)
C
        IF (FREE.GT.0..AND.Z.LT.0.) GO TO 16
C
C*****SEE IF REACHED TIME LIMIT FOR DROP*
        IF (TIME.LT.TLIMIT) GO TO 3
C*****FINISHED FLIGHT
C
        WRITE(9,6905)
6905    FORMAT(/1X'TIME LIMIT EXCEEDED')
        GO TO 6907
16      WRITE(9,6904)
6904    FORMAT(/1X'STORE HAS STRUCK GROUND//')
        IMPACT=2
        GO TO 5430
C
6907    CALL CLOSE(3)
        CALL CLOSE(12)
        CALL CLOSE(13)
        CALL CLOSE(14)
        STOP
        END
```

APPENDIX III

LISTINGS OF TRSET AND TRFREE

```
/TRSET                                         DATE: 16-6-77
/
/ROUTINE TO SET UP THE SKIP CHAIN FOR INTERRUPTS
/FROM THE TRAVERSE RIG READ BUTTON, AND ALSO TO
/SERVICE THAT INTERRUPT.
/
TRSF=705001
/
       .GLOBL TRSET,.DA
/
/CALLING SEQUENCE:   CALL TRSET (IFREE)
/
/
TRSET   XX
        JMS* .DA
        JMP  .+2
IFREE   0
        CAL
        16
        TRSF
        TRBINT
        DZM* IFREE
        JMP* TRSET
/
/
/INTERRUPT SERVICE ROUTINE
/
TRBINT  DAC ACSVE#
        LAC* (0
        DAC PCSVE#
        705002          /CLEAR THE FLAG.
        ISZ* IFREE
        LAC ACSVE
        ION
        JMP* PCSVE
       .END
```

```
C TRFREE                                DATE: 12-7-77
C
C ROUTINE TO CHANGE FROM AN AIRCRAFT-ORIENTED TO AN
C EARTH-AXIS-ORIENTED TRAJECTORY.
C
C OPERATES AT THE POINT WHERE THE STORE IS CONSIDERED TO MOVE
C FROM A TRAJECTORY WHICH IS INFLUENCED BY THE PRESENCE OF
C THE AIRCRAFT TO A UNIFORM-FLOW TRAJECTORY.
C
C
      SUBROUTINE TRFREE (STORCX,HEIGHT)
C
      DIMENSION XTV(3),XV(3),UV(3),C(3,3),B(3,3)
C
      COMMON T(328),IT(5)
C
C ELAPSED TIME
C
      EQUIVALENCE (T(2),TIME)
C
C TIME STEP
C
      EQUIVALENCE (T(3),DTIME)
C
C ORIENTATION MATRIX (BOMB W.R.T. AIRCRAFT)
C
      EQUIVALENCE (T(4),C)
C
C POSITION OF BOMB C.G. W.R.T. AIRCRAFT C.G.
C
      EQUIVALENCE (T(13),X,XV),(T(14),Y),(T(15),Z)
C
C VELOCITY OF BOMB C.G. W.R.T. AIRCRAFT C.G.
C
      EQUIVALENCE (T(16),U,UV),(T(17),V),(T(18),W)
C
C ATTITUDE OF AIRCRAFT W.R.T. EARTH AXES (EULER ANGLES)
C
      EQUIVALENCE (T(76),AROLL),(T(77),APITCH),(T(78),AYAW)
C
C BOMB MASS (KGM)
C
      EQUIVALENCE (T(82),STMASS)
C
C
C GRAVITATIONAL ACCELERATION
C
      EQUIVALENCE (T(92),GRAVAC)
```

```
C
C       FREE STREAM MACH NUMBER
C
        EQUIVALENCE (T(96),RMACH0)
C UNIFORM TRAJECTORY INDICATOR
C       =-1 BEFORE & =+1 AFTER TRANSITION TO BOMB-ALONE TRAJECTORY.
C
        EQUIVALENCE (T(114),FREE)
C
C FREE STREAM VELOCITY
C
        EQUIVALENCE (T(136),VELINF)
C
C GRAVITY FORCES
C
        EQUIVALENCE (T(161),GX),(T(162),GY),(T(163),GZ)
C
C ORIGINAL DIVE ANGLE OF AIRCRAFT
C
        EQUIVALENCE (T(164),DIVE0)
C
C CENTRIPETAL ACCELERATION OF AIRCRAFT (IN G'S)
C
        EQUIVALENCE (T(168),GF)
C
C
C
        EQUIVALENCE (XTV(1),XT),(XTV(2),YT),(XTV(3),ZT)
C
C
C       SET FREE TO +1.
C
        FREE=1.
C
C ADJUST X AND U SO THAT THEY NOW REFER TO EARTH AXES,
C WITH ORIGIN AT GROUND LEVEL BELOW AIRCRAFT C.G. AT TIME OF
C RELEASE (TIME=0).
C
        IF (ABS(GF).GT.1.E-6) GO TO 1
        SV=VELINF*TIME
        VGT=0.
        GO TO 2
1       VG=2.*VELINF/(GF*GRAVAC)
        VGT=TIME/VG
        SV=VG*VELINF*SIN(VGT)
2       DD=DIVE0-VGT
        DIVE=DD-VGT
        CALL TRANS (XTV,AROLL,APITCH,AYAW,-1,XV)
```

```
      X=XT+SV*COS(DD)
      Y=YT
      Z=ZT+HEIGHT
      CALL TRANS (XTV,AROLL,APITCH,AYAW,-1,UV)
      U=XT+VELINF*COS(DIVE)
      V=YT
      W=ZT-VELINF*SIN(DIVE)
C
C REPLACE (C) WITH (C).(CA), SO THAT BOMB
C ORIENTATION IS NOW REFERRED TO EARTH AXES.
C
C NOTE;         C.CA = (CA'.C')'
C  CAN TREAT C' AS 3 COLUMN VECTORS AND USE
C       SUBROUTINE TRANS (DON'T USE CTRANS).
C
      DO31 I=1,3
      DO31 J=1,3
31    B(I,J)=C(J,I)
      DO3 J=1,3
      CALL TRANS (XTV,AROLL,APITCH,AYAW,-1,B(1,J))
      DO3 I=1,3
3     C(J,I)=XTV(I)
C
C SET UP THE NEW DERIVATIVES
C
      CALL DAUX
C
C CLOSE OFF THE 3 PROBE CALIBRATION FILES
C
      CALL CLOSE (3)
      CALL CLOSE (12)
      CALL CLOSE (14)
C
C DEFINE THE STORE COEFFICIENTS FILE
C
      CALL DEFINE (13,29,1664,STORCX,IV4,0,0,0)
C
C SET THE GRAVITY FORCES TO EARTH AXES (CONSTANT)
C
      GX=0.
      GY=0.
      GZ=-STMASS*GRAVAC
C
      WRITE(9,45)
45    FORMAT(///1X'**BOMB-ALONE TRAJECTORY BEGINS**********
      1EARTH AXES FROM THIS POINT**********************************'//)
      RETURN
      END
```

APPENDIX IV

LISTINGS OF ARCTAN, TRANS, CTRANS AND PARAB

```
C ARCTAN                                DATE: 15-6-76
C
      FUNCTION ARCTAN(Y,X)
C FINDS ANGLE WHOSE TANGENT IS Y/X, IN RANGE 0 TO TWO PI.
C
C     CALLED BY MAIN.
      COMMON T(320),IT(5)
      EQUIVALENCE (T(165),PI),(T(166),TWOPI)
C
      ARCTAN=ATAN(Y/ABS(X))
      IF (X.LT.0.) ARCTAN=PI-ARCTAN
      IF (ARCTAN.LT.0.) ARCTAN=ARCTAN+TWOPI
      RETURN
      END
```

```
C TRANS                             DATE: 4-8-76
C
      SUBROUTINE TRANS (XT,ROLL,PITCH,YAW,INV,X)
C
C     CALLED BY MAIN,CALC,CREF,FLOW.
C
C* EULER ANGLES (ROLL,PITCH,YAW) REFER TO NEW AXES W.R.T. OLD AXES.
C* (XT,YT,ZT)=MATRIX (EULER ANGLES)*(X,Y,Z)    IF INV=+1
C* (XT,YT,ZT)= INVERSE MATRIX (EULER ANGLES)*(X,Y,Z)    IF INV=-1
C* MATRIX ELEMENT AIJ=COSINE OF ANGLE BETWEEN NEW I AXIS AND OLD J AXIS.
C* I,J=1,2,3=X,Y,Z.
C  DO NOT ALLOW XT,YT,ZT TO BE THE SAME LOCATIONS IN CALLING ROUTINE
C  AS X,Y,Z,OR IT WILL NOT WORK
C
      DIMENSION C(3,3),A(3,3),X(3),XT(3)
      COMMON T(320),IT(5)
      EQUIVALENCE (T(4),C)
      EQUIVALENCE (A(1,1),A11),(A(1,2),A12),(A(1,3),A13)
      EQUIVALENCE (A(2,1),A21),(A(2,2),A22),(A(2,3),A23)
      EQUIVALENCE (A(3,1),A31),(A(3,2),A32),(A(3,3),A33)
C
C
      SR=SIN(ROLL)
      CR=COS(ROLL)
      SP=SIN(PITCH)
      CP=COS(PITCH)
      SY=SIN(YAW)
      CY=COS(YAW)
C
      A11=CP*CY
      A12=SR*SP*CY+CR*SY
      A13=-CR*SP*CY+SR*SY
      A21=-CP*SY
      A22=-SR*SP*SY+CR*CY
      A23=CR*SP*SY+SR*CY
      A31=SP
      A32=-SR*CP
      A33=CR*CP
      GO TO 2
C
C     ENTRY POINT FOR CTRANS.
C
      ENTRY CTRANS (XT,INV,X)
      DO1 I=1,3
      DO1 J=1,3
1     A(I,J)=C(I,J)
C
2     DO5 I=1,3
      SUM=0.
      DO4 J=1,3
      IF (INV.LT.0) GO TO 3
      SUM=SUM+A(I,J)*X(J)
      GO TO 4
3     SUM=SUM+A(J,I)*X(J)
4     CONTINUE
5     XT(I)=SUM
      RETURN
      END
```

```
C PARAB                                 DATE: 4-8-76
C
      SUBROUTINE PARAB (ABC,X,F)
C
C FITS PARABOLA F(X)=A+B*(X-XA)+C*(X-XA)**2 TO DATA.
C
      DIMENSION ABC(3),X(3),F(3)
C
      DX2=X(2)-X(1)
      DX3=X(3)-X(1)
      ABC(1)=F(1)
      FXA=(F(3)-F(1))/DX3
      ABC(3)=((F(2)-F(1))/DX2-FXA)/(DX2-DX3)
      ABC(2)=FXA-ABC(3)*DX3
      RETURN
      END
```

APPENDIX V

LISTINGS OF FLOW, CALC, TRAV, TRWAIT, ADRD, PROBE AND READIN

```
C FLOW                               DATE: 5-8-76
C
C
      SUBROUTINE FLOW
C
C     CALLS CALC,TRWAIT,ADRD,PROBE,TRANS,CTRANS.
C     CALLED BY MAIN.
C
C*****FINDS FLOW PROPERTIES ALL ALONG BOMB IN BOMB AXES
C*****AND STORES PROPERTIES IN AMS,....WMS.
C
      DIMENSION QMS(12),RMS(12),UMS(12),VMS(12),WMS(12),XMS(12),PRESS(5)
      DIMENSION UPVEC(3),UT(3),UA(3),UVEC(3)
      COMMON T(320),IT(5),NF
      EQUIVALENCE (T(96), RMACH0), (T(99),ATTACK), (T(104), QDP0)
      EQUIVALENCE (T(83),APTROL),(T(84),APTPIT)
      EQUIVALENCE (T(91),VSOUND)
      EQUIVALENCE (T(165), PI)
      EQUIVALENCE (T(193),QMS),(T(205),RMS),(T(217),UMS)
      EQUIVALENCE (T(229),VMS),(T(241),WMS),(T(253),XMS)
      EQUIVALENCE (T(277),DP13,PRESS),(T(278),DP24),
     1 (T(279),PMAN),(T(280),PPIT),(T(281),TPRESS)
C
      EQUIVALENCE (IT(1), NMS)
C
      EQUIVALENCE (UPVEC(1),UP),(UPVEC(2),VP),(UPVEC(3),WP)
      EQUIVALENCE (UT,UVEC),(UPVEC,UA)
C
      DATA TPSTP/116.3497/,TPZERO/1676./,TPRAT/345.3257/
C
C*****STATEMENT FUNCTION FOR TANGENT
      TAN(X)=SIN(X)/COS(X)
C
C DYNAMIC PRESSURE, 1/2 RHO V SQUARED, IS GIVEN BY:
C     QDP0= (GAMMA/2) (RMACH0**2) PSTAT0
C     WHERE PSTAT0= TPRESS/(1+(GAMMA-1)/2RMACH0**2)**(GAMMA/(GAMMA-1))
C     QFACT=QDP0/TPRESS
C
      QFACT=0.7*RMACH0**2/(1.+0.2*RMACH0**2)**3.5
C
      IF(NF.EQ.NMS) GO TO 1
      NS=1
      NF=NMS
      I=NS
      INC=1
      GO TO 2
1     NS=NMS
      NF=1
      I=NS
      INC=-1
C*****FOR EACH POINT ALONG BOMB
```

```
C*****NEXT POINT IS I+INC, UNLESS POINT IS LAST POINT
2       NEXT=I+INC
C*****IF POINT IS LAST POINT, NEXT POINT IS 1, WHICH WILL BE
C*****CLOSE TO FIRST POINT AFTER NEXT INTEGRATION STEP
C*****IS IT THE FIRST POINT?
        IF(I.NE.NS) GO TO 30
C*****CALCULATE 1ST POINT IN BOMB AXES (METRES) IN TRAVERSE AXES
C*****IN INCHES, AND START PROBE MOVING.
        CALL CALC (NS)
C*****WAIT 'TIL PROBE REACHED THIS POINT, AND SETTLED
C*****DOWN SUFFICIENTLY TO TAKE PRESSURE MEASUREMENTS
30      CALL TRWAIT
C*****WHEN READY, TAKE PRESSURE MEASUREMENTS
        CALL ADRD (PMAN,PPIT,DP13,DP24,TPRESS)
        TPRESS=(TPRESS-TPZERO)/TPSTP
        DO84 K=1,5
84      PRESS(K)=PRESS(K)*TPRAT
        QDP0=TPRESS*QFACT
C
        IF(I.EQ.NF) GO TO 67
C*****CALCULATE NEXT POINT IN TRAVERSE AXES (AS BEFORE),
C       AND START PROBE MOVING.
        CALL CALC (NEXT)
C*****CALCULATE FLOW PROPERTIES FROM PRESSURE MEASUREMENTS
67      CALL PROBE (DOWNP,SIDEP,RMS(I),QMS(I))
C*****FIND ACTUAL FLOW VELOCITY COMPONENTS IN PROBE AXES (M/SEC)
        VEL=VSOUND*RMS(I)*SQRT((1.+.2*RMACH0**2)/(1.+.2*RMS(I)**2))
        UP=-VEL/SQRT(1.+TAN(DOWNP)**2+TAN(SIDEP)**2)
        VP=UP*TAN(SIDEP)
        WP=UP*TAN(DOWNP)
C*****FIND VELOCITY IN TUNNEL AXES FROM PROBE AXES
        CALL TRANS (UT,APTROL,APTPIT,0.,-1, UPVEC)
C*****FIND VELOCITY IN PLANE AXES FROM TUNNEL AXES
        CALL TRANS (UA,PI,ATTACK,0.,+1, UT)
C*****FIND VELOCITY IN BOMB AXES FROM PLANE AXES
        CALL CTRANS (UVEC,+1,UA)
        UMS(I)=UVEC(1)
        VMS(I)=UVEC(2)
        WMS(I)=UVEC(3)
C*****CONTINUE FOR ALL MEASUREMENT STATIONS
        IF(I.EQ.NF) RETURN
        I=I+INC
        GO TO 2
        END
```

```
C CALC                      DATE: 4-8-76
C
      SUBROUTINE CALC (J)
C*****CONVERTS POINT (XB,YB,ZB) IN BOMB AXES (METRES) TO
C****POINT (XT,YT,ZT) IN TRAVERSE AXES (INCHES)
C
C     CALLS CTRANS,TRANS.
C     CALLED BY FLOW.
C
      DIMENSION XT(3),XMS(12),XA(3),XI(3)
      COMMON T(320),IT(5)
      EQUIVALENCE (T(13),X),(T(14),Y),(T(15),Z)
      EQUIVALENCE (T(89),SCALE),(T(99),ATTACK)
      EQUIVALENCE (T(165), PI), (T(167), CMI)
      EQUIVALENCE (T(253),XMS)
      EQUIVALENCE (XT,XI)
      EQUIVALENCE (XA(1),XA1),(XA(2),YA),(XA(3),ZA)
C
      XI(1)=XMS(J)
      XI(2)=0.
      XI(3)=0.
C*****TRANSFORM FROM BOMB METRES TO PLANE METRES.
      CALL CTRANS (XA,-1,XI)
C*****THEN CONVERT TO INCHES AND SCALE DOWN TO MODEL SIZE.
      R=CMI/SCALE
      XA1=(XA1+X)*R
      YA=(YA+Y)*R
      ZA=(ZA+Z)*R
C*****THEN TRANSFORM TO TRAVERSE AXES.
      CALL TRANS (XT, PI,ATTACK,0.,-1, XA)
      CALL TRAV (XT)
      RETURN
      END


/TRVCTS                             DATE: 8-3-77
/
/ROUTINE TO MOVE THE TRAVERSE RIG TO A
/POSITION (X,Y,Z) IN THE MINIMUM TIME.
/CALLING SEQUENCE:
/       FORTRAN:   CALL TRAV (X)
/
/       MACRO:  JMS* TRAV
/               JMP .+2
/               400000+X
/
/ASSUMES 'REQUIRED COORDS.' ARE SET PRIOR
/TO ENTRY.
/
TRSX=705101
TRSY=705141
TRSZ=705121
TRSR=705161
/
        .GLOBL TRAV,TRWAIT              /INTERNAL
        .GLOBL .AG,.AK,.AX,.DA          /EXTERNAL
/
TRAV    XX
        JMS* .DA
        JMP .+2
X1      400000+REQXP
INIT    CAL             /SET UP SKIP
```

```
        16                  /CHAIN FOR AN
        TRSX                /INTERRUPT FROM THE
        TRINT               /'X COMPLETE' FLAG
        CAL                 /...
        16                  /...
        TRSY                /'Y COMPLETE' FLAG
        TRINT               /...
        CAL                 /...
        16                  /...
        TRSZ                /'Z COMPLETE' FLAG
        TRINT               /...
        CAL                 /NOTE: THE TRAVERSE
        16                  /SKIP ON ROLL
        TRSR                /MUST BE
        TRINT               /INITIALIZED!!!
        LAC (JMP NSU        /DON'T DO THE
        DAC INIT            /SET-UP AGAIN.
NSU     DZM XFLG            /CLEAR THE THREE
        DZM YFLG            /'COORDINATE MOVING'
        DZM ZFLG            /INDICATORS.
        LAC TRAV            /SET UP THE
        DAC PCSVE#
        705164              /ENABLE TRAVERSE FLAGS.
        LAW -1              /SET THE 'TRAVERSE
        DAC TRIP            /IN PROGRESS' INDICATOR.
        LAW -3              /SET UP TO ACQUIRE
        DAC TRKNT#          /3 COORDINATE VALUES.
        PXA                 /SAVE THE CONTENTS
        DAC XRSVE#          /OF INDEX REGISTER,
        CLX                 /AND CLEAR IT.
        LAC REQXP
        DAC RQPTR
TV1     JMS* .AG            /GET A COORDINATE
RQPTR   XX                  /VALUE X,Y, OR Z.
        ISZ RQPTR
        ISZ RQPTR
        JMS* .AK            /MULTIPLY VALUE BY
        .DSA C1000          /1000 (DECIMAL).
        JMS* .AX            /FIX IT, AND
        DAC XREQ,X          /SAVE BINARY VALUE.
        JMS BINBCD          /CONVERT TO B.C.D.
        DAC RBCD,X          /EQUIVALENT, AND SAVE.
        AXR 1               /BUMP INDEX REGISTER.
        ISZ TRKNT           /ALL 3 COORDS YET?
        JMP TV1             /NO.
        JMP TR1             /YES.
/
REQXP   0
XREQ    .BLOCK 3
RBCD    .BLOCK 3
C1000   12;     372000
XFLG    0       /'COORDINATE MOVING' INDICATORS;
YFLG    0       /NON-ZERO IF THE CORRESPONDING
ZFLG    0       /AXIS IS CURRENTLY IN MOTION.
/
/INTERRUPT HANDLER.
/
TRINT   DAC ACSV2#          /SAVE ACCR CONTENTS.
        LAC* (0             /SAVE INTERRUPT
        DAC PCSV2#          /RETURN ADDRESS.
```

```
        DZM RLFLG#      /TEST WHETHER ANY
        TRSX            /OF THE X,Y, OR Z
        ISZ RLFLG       /TRAVERSE COMPLETE
        TRSY            /FLAGS ARE UP
        ISZ RLFLG       /...
        TRSZ            /...
        ISZ RLFLG       /...
        LAC RLFLG       /'RLFLG' WILL BE
        AAC -3          /3 IF NONE ARE UP
        SZA             /...
        JMP TRPROC      /NOT 3. LEGITIMATE INTERRUPT.
        705104          /NONE ARE UP.
        LAC ACSV2       /MUST HAVE BEEN
        ION             /FROM ROLL AXIS.
        JMP* PCSV2      /THEREFORE IGNORE IT.
/
TRPROC  LAC PCSV2       /SET THE RETURN
        DAC PCSVE#      /POINTER.
        LAC ACSV2       /SET UP TO RESTORE
        DAC ACSVE#      /THE ACCUMULATOR.
        PXA             /SAVE INDEX REGISTER
        DAC XRSVE       /CONTENTS.
TR1     LAW -3          /SET UP TO READ
        DAC TRKNT       /3 COORDINATE VALUES.
        CLX             /CLEAR INDEX REGISTER.
TR2     XCT RDTR,X      /READ BCD COORD VALUE.
        JMS TRBCD       /CONVERT IT TO BINARY.
        TCA             /SUBTRACT 'ACTUAL'
        TAD XREQ,X      /FROM 'REQUIRED'.
        DAC DEL,X       /SAVE THE DIFFERENCE.
        AXR 1           /DO THIS FOR ALL
        ISZ TRKNT       /3 COORDS X,Y,& Z
        JMP TR2         /...
        TRSX            /CLEAR THE
        SKP             /'COORDINATE MOVING'
        DZM XFLG        /INDICATORS OF
        TRSY            /THOSE COORDS FOR
        SKP             /WHICH 'TRAVERSE
        DZM YFLG        /COMPLETE' FLAGS
        TRSZ            /HAVE BEEN
        SKP             /POSTED ..
        DZM ZFLG        /...
        705104          /CLEAR TRAVERSE FLAGS.
        LAW -3
        DAC KNTT#
        CLX
TA1     LAC DEL,X       /TEST THE THREE
        SZA             /'ACTUAL' COORDINATES
        JMP TA2         /X,Y, AND Z FOR
        AXR 1           /EQUALITY WITH
        ISZ KNTT        /THEIR 'REQUIRED'
        JMP TA1         /VALUES.
        JMP TRCOM       /TRAVERSE IS COMPLETE.
TA2     LAW -3          /SET UP TO EXAMINE
        DAC KNTT        /3 COORDINATES.
        CLX             /CLEAR INDEX REGISTER.
TA3     JMS TR          /RELOAD TRAVERSE REGISTER
        AXR 1           /(IF NECESSARY), AND
        ISZ KNTT        /BEGIN TRAVERSING IF
        JMP TA3         /APPROPRIATE.
```

```
TRRTN   LAC XRSVE        /RESTORE THE
        PAX              /INDEX REGISTER,
        LAC ACSVE        /THE ACCUMULATOR,
        ION              /TURN ON INTERRUPT,
        JMP* PCSVE       /AND RETURN.
/
TRCOM   DZM TRIP         /CLEAR THE 'TRAVERSE
        705144           /DISABLE TRAVERSE FLAGS.
        JMS TIME         /SET 'TIM1' TO THE TIME
        AAC 6            /AT WHICH TRAVERSE COMPLETE,
        TCA              /PLUS 6 SECONDS. THIS ALLOWS
        DAC TIM1#        /THE SYSTEM TO SETTLE.
        JMP TRRTN        /RETURN.
/
/ROUTINE TO TEST WHETHER THE TRAVERSE RIG
/INTERFACE REGISTER CAN BE FILLED WITH
/THE 'REQUIRED' VALUE OF A COORDINATE
/AND/OR THE CORRESPONDING COORDINATE
/CAN BE SET TRAVERSING.
/THE ROUTINE EXECUTES WHICHEVER OF THESE
/OPERATIONS IT FINDS TO BE LEGAL.
/
TR      XX
        LAC DEL,X        /RETURN IF THE
        SNA              /'ACTUAL' VALUE EQUALS
        JMP* TR          /THE 'REQUIRED' VALUE.
        DZM DREGR
        DZM TROK
        LAC (LAC XFLG    /SET THE SWITCHES
        DAC TST1         /IN SUBROUTINE 'TEST'
        LAC (LAC XREQ    /SO THAT THEY
        DAC TST2         /WILL EXAMINE THE
        LAC (TAD DEL     /'X' COORDINATE
        DAC TST3         /...
        LAW -3           /SET UP TO EXAMINE
        DAC TRKNT        /3 COORDINATES.
TRB1    JMS TEST
        ISZ TST1         /BUMP THE SWITCHES
        ISZ TST2         /IN 'TEST' TO
        ISZ TST3         /THE NEXT COORD.
        ISZ TRKNT        /3 COORDS YET?
        JMP TRB1         /NO.
        LAC DREGR        /O.K. TO RELOAD
        SZA              /TRAVERSE REGISTER.
        JMP TRB2         /NO.
        LAC RBCD,X       /RELOAD REGISTER IF
        JMS LDREG        /'DREGR' IS ZERO.
TRB2    LAC TROK         /O.K. TO TRAVERSE
        SZA              /THIS COORDINATE?
        JMP* TR          /NO.
        ISZ XFLG,X       /YES. SET INDICATOR
        XCT ITR,X        /AND INITIATE TRAVERSE.
        JMP* TR          /RETURN.
/
/ROUTINE TO CARRY OUT THREE TESTS AND
/SET LOCATIONS 'DREGR' AND 'TROK'
/ACCORDINGLY.
/UPON RETURN FROM 'TEST',
/       (A)  THE TRAVERSE INTERFACE REGISTER CAN BE
/            RELOADED IF, & ONLY IF, DREGR=0.
```

```
/       (8)  THE COORDINATE XREQ,X CAN BE SET
/            TRAVERSING IF, & ONLY IF, TROK=0.
/
/
TEST    XX
TST1    LAC XFLG          /RETURN IF THE
        SNA               /COORDINATE IS
        JMP* TEST         /NOT MOVING.
TST2    LAC XREQ          /RETURN IF 'REQUIRED'
        TCA               /VALUES OF THE
        TAD XREQ,X        /MOVING COORD
        DAC TSTSV#        /AND THAT OF XREQ,X
        SNA               /ARE EQUAL
        JMP* TEST         /...
TST3    TAD DEL           /RETURN IF XREQ,X
        XOR TSTSV         /IS BETWEEN THE
        SPA               /MOVING COORD AND
        JMP* TEST         /ITS 'REQUIRED' VALUE.
        ISZ DREGR         /DON'T LOAD REGISTER.
        LAC XREQ,X        /IS THE CURRENT
        TCA               /VALUE OF THE
        TAD BREGR         /TRAVERSE REGISTER
        DAC TSTSV         /BETWEEN THE 'REQUIRED'
        TAD DEL,X         /COORD XREQ,X
        XOR TSTSV         /AND ITS 'ACTUAL'
        SMA               /VALUE?
        ISZ TROK          /NO. DON'T TRAVERSE.
        JMP* TEST         /YES. RETURN.
DREGR   XX
TROK    XX
/
TRIP    0       /'TRAVERSE IN PROGRESS' INDICATOR.
TRUND   0
ITR     705102  /INITIATE TRAVERSE, X AXIS.
        705142  /INITIATE TRAVERSE, Y AXIS.
        705122  /INITIATE TRAVERSE, Z AXIS.
RDTR    705012  /READ POSITION, X AXIS.
        705052  /READ POSITION, Y AXIS.
        705032  /READ POSITION, Z AXIS.
DEL     0;      0;      0
/
/
DELAY   XX
        DAC DLYKNT#
        ISZ DLYKNT
        JMP .-1
        JMP* DELAY
LDREG   XX
        DAC REGR#         /SAVE THE BCD.
        JMS TRBCD         /CONVERT TO BINARY,
        DAC BREGR#        /AND SAVE IT.
        LAW -6065         /10 MILLISECOND
        JMS DELAY         /SETTLING DELAY.
        LAC REGR          /LOAD BCD INTO
        705004            /TRAVERSE INTERFACE REGISTER.
        LAW -5            /WAIT FOR REGISTER
        JMS DELAY         /TO SETTLE (20 MICROSECS),
        JMP* LDREG        /AND RETURN.
/
/ROUTINE TO CONVERT TO BINARY
/FROM TRAVERSE RIG BCD:
```

```
/                                X X XXXX XXXX XXXX XXXX
/                                S 1 8421 8421 8421 8421
/
TRBCD   XX
        DAC BCD#
        DZM BIN#
        RTL
        DAC TEMP#
        SZL
        ISZ BIN
        LAW -4
        DAC KNT#
BC1     LAC BIN
        CLL
        MUL
        12
        LACQ
        DAC BIN
        LAC TEMP
        RTL
        RTL
        DAC TEMP
        RAL
        AND (17
        TAD BIN
        DAC BIN
        ISZ KNT
        JMP BC1
        LAC BCD
        RAL
        LAC BIN
        SPL
        TCA
        JMP* TRBCD
/
/ROUTINE TO CONVERT BINARY
/TO 8421 BCD.
/CALLING SEQUENCE:
/WITH THE BINARY IN THE AC,
        /JMS* BINBCD
        /NORMAL RETURN
/BCD IS RETURNED IN THE AC.
/
BINBCD  XX
        DAC BBIN#
        DZM BCD#
        DZM SIGN#
        SMA
        JMP BB1
        TCA
        DAC BBIN
        LAC (400000
        DAC SIGN
BB1     LAW -5
        DAC BKNT#
        LAC BBIN
        LMQ
BB2     SPL!CLL!CLA
        LAC (040000
        DAC LINK#
```

```
        CLA
        DIV
        12
        XOR BCD
        RTR
        RTR
        XOR LINK
        DAC BCD
        ISZ BKNT
        JMP BB2
        RAL
        XOR SIGN
        JMP* BINBCD
/
/ROUTINE TO READ THE MACRO TIME FROM THE REAL TIME
/CLOCK, AND CONVERT TO SECONDS AFTER MIDNIGHT.
/
/CALLING SEQUENCE:
/       JMS TIME
/       NORMAL RETURN
/TIME IS RETURNED IN THE AC.
/
TIME    XX
        702512
        CMA
        JMS TAUX
        JMS MUL60
        DAC TRTIME
        702552
        CMA
        DAC MINSEC
        RCL
        SWHA
        AND (377
        JMS TAUX
        TAD TRTIME
        JMS MUL60
        DAC TRTIME
        LAC MINSEC
        AND (377
        JMS TAUX
        TAD TRTIME
        JMP* TIME
/
TRTIME  0
MINSEC  0
/
/
TAUX    XX
        DAC TEMPX
        RTR
        RTR
        AND (17
        CLL
        MUL
        12
        LACQ
        DAC TEMPX+1
        LAC TEMPX
        AND (17
```

```
        TAD TEMPX+1
        JMP* TAUX
/
TEMPX   0;      0
/
/
MUL60   XX
        CLL
        MUL
        74
        LACQ
        JMP* MUL60
/
/ROUTINE TO WAIT AT LEAST 6 SECONDS FROM THE TIME
/OF COMPLETION OF THE PREVIOUS TRAVERSE.
/ THE CURRENT TIME IS READ FROM THE REAL TIME CLOCK
/ AND IS COMPARED WITH THE CONTENTS OF LOCATION
/'TIM1', WHICH WAS SET AT 'TRAVERSE COMPLETE'.
/THE PROGRAM IS TRAPPED UNTIL BOTH THE 'TRAVERSE IN PROGRESS'
/INDICATOR (TRIP) IS CLEAR AND THE CURRENT TIME IS GREATER
/THAN THE CONTENTS OF TIM1.
/
/CALLING SEQUENCE:
/       JMS* TRWAIT
/       NORMAL RETURN
/
TRWAIT  XX
TW1     LAC TRIP
        SZA
        JMP TW1
        LAC TIM1
        SNA
        JMP* TRWAIT
TW2     JMS TIME
        TAD TIM1
        SPA
        JMP TW2
        DZM TIM1
        JMP* TRWAIT
/
        .END


/ADRD                           DATE: 21-4-76
/
/ROUTINE TO READ THE
/
/               MANIFOLD PRESSURE
/               PITOT PRESSURE
/               DIFFERENTIAL PRESSURES DP13 & DP24
/
/FROM THE RAYTHEON MULTIVERTER,
/AND THE S1 TOTAL PRESSURE FROM THE DIGITIZER.
/
/CALLING SEQUENCE:   CALL ADRD (PMAN,PPIT,DP13,DP24,TPRESS)
/
/CALLED BY FLOW
/
/
/
/
/
```

```
DT=1
AD=6
/
        .IODEV DT,AD
/
        .GLOBL ADRD
        .GLOBL .DA,.AG,.AH,.AI,.AK,.AL,.AN,.AW
        .GLOBL CPBIN
/
ADRD    XX
        JMS* .DA
        JMP .+6
PM      0
PP      0
P13     0
P24     0
TP      0
/
ONCE    .INIT AD,0,ADRD
        .INIT DT,0,ADRD
        .SEEK DT,COMPBL
        .READ DT,0,AMPVEX,250
        .WAIT DT
        .CLOSE DT
        LAC VEX1P3
        TAD (TAD ADBUF
        DAC ADDVEX
        LAC (JMP A1
        DAC ONCE
/
A1      .READ AD,0,ADBUF,16
        .WAIT AD
        LAC ADBUF+22
        TCA
ADDVEX  XX
        JMS* .AW
        JMS* .AN
        C5
        JMS* .AH
        VEX5
        PXA
        DAC XRSVE#
        PLA
        DAC LSVE#
        CLX
        LAC (4
        PAL
        LAC (AMPGNS
        DAC GAINP
        LAC (PCAL
        DAC PCONS
A2      LAC PM,X
        DAC PPTR
        LAC AMPBL,X
        TAD (TAD ADBUF
        DAC ADDCH
        LAC AMPZER,X
        TCA
ADDCH   XX
        JMS* .AW
```

```
        JMS* .AL
GAINP   XX
        ISZ GAINP
        ISZ GAINP
        JMS* .AK
        VEX5
        JMS* .AH
        TEMP
        JMS POLY
PCONS   XX
        TEMP
PPTR    XX
        LAC PCONS
        AAC 6
        DAC PCONS
        AXS 1
        JMP A2
/
/READ TOTAL PRESSURE
/
        703412          /READ S1TP.
        JMS* CPBIN
        JMS* .AW
        LAC TP
        DAC TPTR
        JMS* .AH
TPTR    XX
        LAC (5
        PAL
ZARG    DZM PM,X
        AXS 1
        JMP ZARG
        LAC XRSVE
        PAX
        LAC LSVE
        PAL
        JMP* ADRD
/
/DATA AREA:
/
ADBUF   .BLOCK 23
/
TEMP    0;      0
VEX5    0;      0
C5      3;      240000  /5.
COMPBL  .SIXBT 'COMPBLDAQ'
/
/DATA INFORMATION BLOCK:
/
AMPVEX  174000; 0
DATSCE  .BLOCK 20
AMPBL   .BLOCK 12
NAMP    0
VEX1P3  0
CHKNT   0
GP2PR   0
AMB     .BLOCK 20
VEXBL   .BLOCK 20
SYSREQ  .BLOCK 5
MACREQ  0
```

```
AMPGNS   .BLOCK 24
AMPZER   .BLOCK 12
PCAL     .BLOCK 74
THERMS   .BLOCK 6
THERMC   .BLOCK 6
BALK     .BLOCK 110
CPRES    .BLOCK 4
/
/
/CALLING SEQUENCE:
/        JMS POLY
/        PTR TO CAL CONSTS.
/        PTR TO VOLTAGE
/        PTR TO PRESSURE RETURN.
/        NORMAL RETURN.
/
POLY     XX
         LAC* POLY
         ISZ POLY
         DAC PL2
         AAC 2
         DAC PL3
         AAC 2
         DAC PL5
         LAC* POLY
         ISZ POLY
         DAC PL1
         DAC PL4
         LAC* POLY
         ISZ POLY
         DAC PL6
         JMS* .AG
PL1      XX                 /V
         JMS* .AK
PL2      XX                 /A*V
         JMS* .AI
PL3      XX                 /A*V+B
         JMS* .AK
PL4      XX                 /(A*V+B)*V
         JMS* .AI
PL5      XX                 /A.V2+B.V+C
         JMS* .AH
PL6      XX
         JMP* POLY
/
         .END ADRD


C PROBE                                  DATE: 31-10-77
C
C
C       USES DIRECT ACCESS I/O.
C
        SUBROUTINE PROBE (DOWNP,SIDEP,RMACH,Q)
C
C       CALLS READIN.
C       CALLED BY FLOW.
C
        DIMENSION DOWNS(2),SIDES(2),ANGLE(2)
        COMMON T(320),IT(5)
```

```
      EQUIVALENCE (T(134),DTR),(T(135),RTD)
      EQUIVALENCE (T(277),DP13),(T(278),DP24)
      EQUIVALENCE (T(279),PMAN),(T(280),PPIT),(T(281),TPRESS)
C
      EQUIVALENCE (ANGLE(1),DOWN),(ANGLE(2),SIDE)
C
      DATA ERROR/0.15/,PITCHN/0./
C
      TAN(A)=SIN(A)/COS(A)
C
C*****FIND PRESSURE RATIO
      PRATIO=PPIT/PMAN
      IF (PRATIO.GT.1.) GO TO 43
      WRITE(4,42) PPIT,PMAN
42    FORMAT(/1X'(PROBE) PITOT=',G12.5,' < MANIFOLD=' G12.5)
      STOP 5
C*****SET PITCH=LAST PITCH FOUND (I.E.,AT LAST MEASUREMENT
C*****STATION,WHICH VALUE WILL BE CLOSE TO THIS ONE)
43    PITCH=PITCHN
C*****GO THRU MAX OF 5 TIMES
      NTHRU=0
      NTMAX1=9
C*****FIND MACH NUMBER AT THIS RATIO,PITCH
10    IF (NTHRU.LE.4) GO TO 44
      WRITE(4,45) PITCH,DIFF,PRATIO,RMACH,DOWN,SIDE
45    FORMAT(/1X'(PROBE) 5 ITERATIONS EXCEEDED'/
      1 1X'PITCH,DIFF,PRATIO,RMACH,DOWN,SIDE'/
      2 1X6G12.5)
      STOP 11
44    IR=65*IFIX(PITCH)+IFIX(20.*(PRATIO-1.))+1
      CALL READIN(14,RMACH,1,2,PRATIO,PITCH,65,IR)
      AM1=RMACH*10.
      M1=AM1-3.
      IF (M1.GE.1.AND.M1.LE.11) GO TO 53
      WRITE(4,52) RMACH
52    FORMAT(/1X'(PROBE) MACH NO. = ',G12.5,' OUTSIDE RANGE')
      STOP 12
C*****FIND TOTAL PRESSURE AT THIS MACH,PITCH
53    IR=11*IFIX(PITCH)+M1
      CALL READIN(3,RATIO,1,2,RMACH,PITCH,11,IR)
      PTOT=PPIT/RATIO
C*****FIND STATIC PRESSURE
      PSTAT=PTOT/((1.+0.2*RMACH**2)**3.5)
C*****FIND DYNAMIC PRESSURE-
      Q=0.5*1.4*PSTAT*RMACH**2
C*****FIND COEFFICIENTS
      DC13=DP13/Q
      DC24=DP24/Q
C*****FIND MACH NUMBER INTEGERS ON EACH SIDE OF ACTUAL MACH NO
C*****FOR EACH MACH,FIND DOWN-AND SIDE-WASH ANGLES (DEGREES)
      IR=((M1-1)*41+IFIX(10.*(DC24+2.)))*41+IFIX(10.*(DC13+2.))+1
      IF(IR.LE.10*1681) GO TO 47
      WRITE(4,46) IR,RMACH,DC13,DC24
46    FORMAT(/1X'(PROBE) RECORD NO. =',I6,' EXCEEDS 11*1681'/
      1 1X'RMACH,DC13,DC24:  ',3G15.6)
      STOP 1
47    DO20 M=1,2
      CALL READIN(12,ANGLE,2,2,DC13,DC24,41,IR)
      DOWNS(M)=DOWN
      SIDES(M)=SIDE
```

```
20      IR=IR+1681
C*****LINEAR INTERPOLATION BETWEEN MACH NUMBERS
        FMACH=M1
        FMACH=AM1-FMACH-3.
        DOWNP=DOWNS(1)+(DOWNS(2)-DOWNS(1))*FMACH
        SIDEP=SIDES(1)+(SIDES(2)-SIDES(1))*FMACH
C*****CONVERT PROBE ANGLES TO RADIANS
        DOWNP=DOWNP*DTR
        SIDEP=SIDEP*DTR
C*****CALCULATE NEW PITCH IN DEGREES
        PITCHN=ATAN (SQRT (TAN (DOWNP)**2+TAN (SIDEP)**2))*RTD
C*****ABSOLUTE DIFFERENCE BETWEEN NEW AND OLD PITCH
        DIFF=ABS (PITCHN-PITCH)
C*****SET OLD PITCH EQUAL NEW PITCH-
        PITCH=PITCHN
C*****INCREASE NO OF TIMES GONE THRU
        NTHRU=NTHRU+1
C*****COMPARE WITH ERROR SET ABOVE
        IF (DIFF.GT.ERROR) GO TO 10
C*****RETURN WITH PROBE ANGLES IN RADIANS
        RETURN
        END


C READIN                                        DATE: 15-7-76
C
C
        SUBROUTINE READIN (NDAT,F,NF,IND,X,Y,IXDIM,IR)
C
C       CALLED BY CREF,PROBE.
C
C*****THIS SUBROUTINE READS TABULATED FUNCTIONS FROM DISK
C*****FROM .DAT+2,AND PERFORMS A 2-D LINEAR INTERPOLATION TO FIND THE
C*****VALUES OF THE FUNCTIONS AT THE POINT (X,Y).
C*****CALL IT WITH THE FOLLOWING PARAMETERS...
C*****F(1) ... F(NF)= UP TO 10 INTERPOLATED FUNCTION VALUES (RETURNED).
C*****NF=NUMBER OF FUNCTIONS TO BE INTERPOLATED.
C*****IND=NO OF INDEPENDENT VARIABLES IN RECORD (REAL + INTEGER)
C*****X,Y=POINT AT WHICH WANT INTERPOLATED VALUE OF FUNCTION.
C*****DATA STRUCTURE ON DISK ASSUMED TO BE OF THE FORM...
C*****1ST RECORD - NX,NY (DIMENSIONS OF X,Y TABLES)
C*****2ND RECORD - (XT(I),I=1,NX) (YT(I),I=1,NY)  TABULATED VALUES.
C*****NEXT NX*NY RECORDS - EACH OF THE FORM
C*****IX IY (IZ) XT(IX) YT(IY) (ZT(IZ)) F(1) .... F(NF)
C*****WITH IX VARYING MOST RAPIDLY,THEN IY (THEN IZ,IF PRESENT).
C*****INTERPOLATES BY USING 2X2 POINTS ABOUT ACTUAL ONE.
C*****I.E.,THE 4 TABULATED POINTS CLOSEST TO ACTUAL POINT (X,Y).
C
C*****X,Y TABLES (MAXIMUM OF 65 VALUES AT PRESENT)
        DIMENSION XT(2,2)
C*****UP TO 10 FUNCTION VALUES CAN BE READ IN
C*****UP TO 3 INTEGER AND REAL NUMBERS IX,IY,IZ,XT(IX),YT(IY),ZT(IZ)
        DIMENSION IT(3),RT(3)
C*****2X2=4 POINTS FOR EACH OF UP TO 10 FUNCTIONS WANTED FOR INTERP
        DIMENSION FW(10,2,2)
C*****THE INTERPOLATED VALUES (UP TO 10) ARE RETURNED THRU F
        DIMENSION F(1)
        COMMON T(320),IIT(5)
        EQUIVALENCE (T(135),RTD)
```

```
        DO100 I=1,2
        DO100 J=1,2
        IRN=IR+IXDIM*(J-1)+(I-1)
        READ(NDAT'IRN)(IT(L),L=1,IND),(RT(L),L=1,IND),
        1 (FW(L,I,J),L=1,NF)
C       IF(NDAT.EQ.13)WRITE(9,301) (FW(L,I,J),L=1,NF),NDAT,IRN,
C       1 (IT(L),L=1,IND)
C 301   FORMAT(1X'FW (R/IH)'/1X8G15.6/1X2G15.6,'  NDAT,IRN,IT   ',10I6)
        JK=2-IABS(I-J)
100     XT(I,J)=RT(JK)
C       IF(NDAT.EQ.13)WRITE(9,300) ((XT(I,J),J=1,2),I=1,2)
C 300   FORMAT(1X'XT (READIN)',4G15.6)
C*****CARRY OUT 2-D INTERPOLATION
C*****FOR EACH FUNCTION
        DX=(X-XT(1,2))/(XT(2,1)-XT(1,2))
        DY=(Y-XT(1,1))/(XT(2,2)-XT(1,1))
        DO120 N=1,NF
C*****2-D INTERPOLATION,AND RETURN RESULT THRU F
C*****FOR THE POINT FW(N,I,J)
        F11=FW(N,1,1)
        F12=FW(N,1,2)
        F21=FW(N,2,1)
        F22=FW(N,2,2)
120     F(N)=F11*(1.-DX-DY)+F21*DX+F12*DY+(F11+F22-F12-F21)*DX*DY
        RETURN
        END
```

APPENDIX VI

LISTINGS OF CORECT AND CREF

```
C CORECT                                  DATE: 26-7-76
C
      SUBROUTINE CORECT (ALPHA,W)
C
C        CALLS PARAB.
C        CALLED BY MAIN.
C
C*    THIS ROUTINE CONSTRUCTS THE STREAMLINE ABOUT THE REFERENCE POINT AND
C*    THUS CALCULATES THE CHORD ANGLE ALPHA AND CORRECTION ANGLE GAMMA
C*    THIS CONSTRUCTION IN THE Z-X PLANE IS BASED ON THIS ANALYSIS- (RADIAN
C*    SLOPE OF STREAMLINE AT POINT X = DZ/DX = (DZ/DT)/(DX/DT)
C*    = W(X)/U(X) (IN TERMS OF STREAM VELOCITY COMPONENTS).
C*    HENCE THE STREAMLINE IS DEFINED BY Z(X)=INTEGRAL FROM ZERO TO X
C*    OF W(X)/U(X).
C
        DIMENSION U(12),W(1),X(12),XV(3),F(3),ABC(3)
        COMMON T(328),IT(5)
        EQUIVALENCE (IT(2), NREF)
        EQUIVALENCE (T(217), U), (T(253), X)
        EQUIVALENCE (ABC(1),A),(ABC(2),B),(ABC(3),C)
        EQUIVALENCE (XV(1),XA),(XV(2),XB),(XV(3),XC)
C
C     *FIND THREE POINTS ABOUT THE REFERENCE POINT*
        IXA=NREF-1
        IF (NREF.EQ.1) IXA=1
        IX=IXA
C     *DEFINE POINTS XA,XB,XC ABOUT MEASUREMENT STATION XB*
C
C       INTEGRAND F=W/U
C
        DO1 L=1,3
        XV(L)=X(IX)-X(IXA)
        F(L)=W(IX)/U(IX)
1       IX=IX+1
C     *FIT PARABOLA F=A+B*X+C*X**2 TO THE INTEGRAND*
        CALL PARAB (ABC,XV,F)
C     *INTEGRATE IT FROM ZERO TO EACH POINT XA,XB,XC TO OBTAIN
C     *STREAMLINE POINTS ZA,ZB,ZC, THUS - Z=A*X+B*X**2/2+C*X**3/3.*
        ZBXB=(C/3.*XB+B/2.)*XB+A
        ZCXC=(C/3.*XC+B/2.)*XC+A
C     *STRAIGHT-LINE CHORD DEFINED BY Z=(ZC/XC)*X.*
C     *DIFFERENCE BETWEEN CHORD AND STREAMLINE AT REFERENCE POINT*
        DIFF=(ZBXB-ZCXC)*XB
C*****ALPHA IS CHORD ANGLE, TWOGAM IS CORRECTION ANGLE.
        ALPHA=ATAN(ZCXC)
        TWOGAM=2.*DIFF/XC
C
C       CALCULATE EFFECTIVE INCIDENCE
C
        ALPHA=ALPHA+TWOGAM
        RETURN
        END
```

```
C CREF                               DATE: 1-7-77
C
C
        SUBROUTINE CREF
C
C       CALLS TRANS,READIN.
C       CALLED BY MAIN.                                               (SYSTEM
C*    COMPUTES FORCE AND MOMENT COEFFICIENTS DUE TO WIND,IN TOTAL AXIS
C*    BY INTERPOLATION INTO DATA TABLES
C*    PITCHT (0 2 4...30)   ROLLT (-45 -37.5...45)
C*****RMACHT (.4,.5,.6,.7,.8,.85,.9,.95)
C*****THE COEFFICIENTS ARE ON DISK
C*****EACH RECORD WITHIN A FILE CONTAINING
C*****I,J,K,PITCH,ROLL,MACH,CX,CY,CZ,CL,CM,CN,CLP,CMQ,CNP,CYP.
C
        DIMENSION C(10),CINT(2,10),PTVEC(3)
C***** (CINT(M,I) CONTAINS COEFF I AT TWO MACH NOS M)
        DIMENSION UMS(12),VMS(12),WMS(12),RMS(12)
        DIMENSION RMACHT(8)
        COMMON T(320),IT(5)
        EQUIVALENCE (T(94),THETOT),(T(95),PHITOT)
        EQUIVALENCE (T(98),DMAX)
        EQUIVALENCE (T(129),P),(T(130),Q),(T(131),R)
        EQUIVALENCE (T(134),DTR),(T(135),RTD)
        EQUIVALENCE (T(165),PI)
        EQUIVALENCE (T(205),RMS)
        EQUIVALENCE (T(217),UMS),(T(229),VMS),(T(241),WMS)
        EQUIVALENCE (T(265),CX),(T(266),CY),(T(267),CZ)
        EQUIVALENCE (T(268),CL),(T(269),CM),(T(270),CN)
        EQUIVALENCE (IT(2),NREF)
        EQUIVALENCE (C(1),TCX),(C(2),TCY),(C(3),TCZ)
        EQUIVALENCE (C(4),TCL),(C(5),TCM),(C(6),TCN)
        EQUIVALENCE (C(7),TCLP),(C(8),TCMQ,TCNR),(C(9),TCNP)
        EQUIVALENCE (C(10),TCYP)
        EQUIVALENCE (PTVEC(1),PT),(PTVEC(2),QT),(PTVEC(3),RT)
C*****MACH NUMBERS AT WHICH COEFFS ARE TABULATED
        DATA RMACHT/.4,.5,.6,.7,.8,.85,.9,.95/
C
        THDEG=THETOT*RTD
C
        IF(THDEG.GE.0..AND.THDEG.LE.30.) GO TO 3
        WRITE(4,232) THDEG
232     FORMAT(/1X'THETOT =' F12.2,' OUTSIDE RANGE')
        STOP 3
3       PHIDEG=PHITOT*RTD
C*****GET ROLL (PHI) FROM (0 TO 360)  INTO RANGE OF TABLES (-45 TO +45)
C
        IPHI=(PHIDEG+45.)/90.
        PHIDEG=PHIDEG-90.*FLOAT(IPHI)
C*****FIND MACH NUMBER INTEGERS ON EITHER SIDE OF ACTUAL ONE
        RMSREF=RMS(NREF)
        DO1 MACH=1,8
        IF(RMSREF.GE.RMACHT(MACH).AND.RMSREF.LT.RMACHT(MACH+1))GO TO 2
1       CONTINUE
```

```
         WRITE(4,21) RMSREF
21       FORMAT(/1X'(CREF) MACH NO.= ',G12.5,' OUTSIDE RANGE')
C     *EXPRESS EULER RATES IN TOTAL AXIS SYSTEM*
2        CALL TRANS (PT,PHITOT,0.,0.,-1,P)
C*****FOR EACH MACH NUMBER (MACH AND MACH+1)
         IR=(MACH-1)*208+IFIX((PHIDEG+45.)/7.5)*16+IFIX(THDEG/2.)+1
         DO600 N=1,2
C        WRITE(9,4291) IR,MACH,PHIDEG,THDEG
C 4291   FORMAT(1X'IR,MACH,PHIDEG,THDEG',I6,I6,2G15.6)
C*****INTERPOLATE TO GET 10 COEFFS AT THIS PITCH,ROLL,MACH.
         CALL READIN(13,C,10,3,THDEG,PHIDEG,16,IR)
C*****STORE 10 COEFFS AT THIS MACH NO.
         DO 10 I=1,10
   10 CINT(N,I)=C(I)
C*****DO FOR BOTH MACH NUMBERS,EACH SIDE OF ACTUAL ONE
600      IR=IR+208
C*****NOW INTERPOLATE BETWEEN MACH NUMBERS TO GET FINAL COEFFS
         RMACH1=RMACHT(MACH)
         RMACH2=RMACHT(MACH+1)
         DO 500 I=1,10
         SLOPE=(CINT(2,I)-CINT(1,I))/(RMACH2-RMACH1)
500      C(I)=CINT(1,I)+(RMSREF-RMACH1)*SLOPE
C     *ADD COMPONENT PARTS TO GIVE RESULTANT COEFFICIENTS*
         VELREF=SQRT(UMS(NREF)**2+VMS(NREF)**2+WMS(NREF)**2)
C        TCX=TCX
         TCY=TCY+TCYP*PT*DMAX/2./VELREF
C        TCZ=TCZ
         TCL=TCL+TCLP*PT*DMAX/2./VELREF
         TCM=TCM+TCMQ*QT*DMAX/2./VELREF
         TCN=TCN+(TCNP*PT-TCNR*RT)*DMAX/2./VELREF
C*****TRANSFORM TO BOMB AXES FROM TOTAL
         CALL TRANS (CX,PHITOT,0.,0.,+1,TCX)
         CALL TRANS (CL,PHITOT,0.,0.,+1,TCL)
         RETURN
         END
```

APPENDIX VII

LISTING OF CINC

```
C CINC                              DATE: 26-8-76
C
      SUBROUTINE CINC
C
C     CALLED BY MAIN.
C
C*    CALCULATES INCREMENTS IN COEFFICIENTS DUE TO FLOW VARIATION ALONG BOMB
C*****IN BOMB AXES.
C
      DIMENSION ASTOR(12),D(12),Q(12),R(12),U(12),V(12),W(12),X(12)
      DIMENSION FYL(12),FZL(12),FML(12),FNL(12),FYZ(12,2),DF(2,2)
      DIMENSION DADX(12),DVDX(12),DWDX(12),DDX(12,3)
      DIMENSION ABC(3)
      COMMON T(320),IT(5)
      EQUIVALENCE (T(91),VS),(T(97),AMAX),(T(98),DMAX),(T(104),QDP0)
      EQUIVALENCE (T(96),RMACH0),(T(136),VELINF)
      EQUIVALENCE (T(169),ASTOR),(T(181),D),(T(193),Q)
      EQUIVALENCE (T(217),U),(T(229),V),(T(241),W)
      EQUIVALENCE (T(253),X)
      EQUIVALENCE (T(272),DCY),(T(273),DCZ),(T(275),DCM),(T(276),DCN)
C
      EQUIVALENCE (IT(1),NMS),(IT(2),NREF),(IT(3),NB),(IT(4),NF)
C
      EQUIVALENCE (DADX,DDX(1,1)),(DVDX,DDX(1,2)),(DWDX,DDX(1,3))
      EQUIVALENCE (ABC(1),A),(ABC(2),B),(ABC(3),C)
      EQUIVALENCE (FYZ(1,1),FYL,FNL),(FYZ(1,2),FZL,FML)
C
      DATA ETACDC/0.864/
C
      SUMF(DX)=((C*DX/3.+B/2.)*DX+A)*DX
      SCSF(ANG)=SIN(2.*ANG)*COS(ANG/2.)
C
      NFM2=NF-2
C
C FIND DA/DX, DV/DX AND DW/DX.
C METHOD:
C     IF F = A + B(X-XA) + C(X-XA)**2,
C     THEN DF/DX = B + 2C(X-XA).
C
C     IF THE DERIVATIVE IS REQUIRED AT POINT J, THEN THE
C ROUTINE FITS PARABOLAS SUCCESSIVELY THROUGH THE SETS OF POINTS
C (J-2,J-1,J), (J-1,J,J+1) AND (J,J+1,J+2),  TO OBTAIN
C 3 ESTIMATES OF DF/DX BY THE FORMULA ABOVE.
C     PROGRAM THEN AVERAGES THE THREE ESTIMATES.  AT THE END
C POINTS, THE TREATMENT IS SLIGHTLY DIFFERENT.
C
      DO4 L=1,3
      DO1 I=1,NMS
1     DDX(I,L)=0.
      NBM2=NMS-2
      DO2 I=1,NBM2
      XA=X(I)
      GO TO (9,10,11), L
9     CALL PARAB (ABC,X(I),ASTOR(I))
```

```
      GO TO 12
10    CALL PARAB (ABC,X(I),V(I))
      GO TO 12
11    CALL PARAB (ABC,X(I),W(I))
12    DDX(I,L)=DDX(I,L)+B
      I1=I+1
      DDX(I1,L)=DDX(I1,L)+B+2.*C*(X(I1)-XA)
      I2=I+2
2     DDX(I2,L)=DDX(I2,L)+B+2.*C*(X(I2)-XA)
      IF (NMS.EQ.3) GO TO 4
      DDX(2,L)=DDX(2,L)/2.
      DDX(NMS-1,L)=DDX(NMS-1,L)/2.
      IF (NMS.EQ.4) GO TO 4
      DO3 I=3,NBM2
3     DDX(I,L)=DDX(I,L)/3.
4     CONTINUE
C
C     COMPUTE VALUES AT REFERENCE POINT.
C
      UR=U(NREF)
      VR=V(NREF)
      WR=W(NREF)
      QR=Q(NREF)
C
C     VISCOUS FORCE/D(I) = 1/2 RHO ETA CDC V SQRT(V**2+W**2)
C WHERE RHO = 2Q/(U**2+V**2+W**2).
C
      R1=QR*ETACDC*SQRT(VR**2+WR**2)/(UR**2+VR**2+WR**2)
      VYR=R1*VR
      VZR=R1*WR
C
C     SLENDER BODY FORCE/(DA/DX) = Q SIN(2 ANG) COS(ANG/2)
C WHERE ANG = ARCTAN(W/U) FOR FY
C AND       = ARCTAN(V/U) FOR FZ.
C
      ALPHA=ATAN(WR/UR)
      BETA=ATAN(VR/UR)
      QALPH=QR*SCSF(ALPHA)
      QBET=QR*SCSF(BETA)
C
C     AT EACH MEASUREMENT STATION, CALCULATE THE
C     CURVED FLOW INCREMENTS.
C
      DO5 I=1,NMS
      QI=Q(I)
      UI=U(I)
      VI=V(I)
      WI=W(I)
      RHO=2.*QI/(UI**2+VI**2+WI**2)
C
C     SLENDER BODY INCREMENTS.
C
      ALPHA=ATAN(WI/UI)
      BETA=ATAN(VI/UI)
      DFYSB=(QI*SCSF(BETA)-QBET)*DADX(I)
      DFZSB=(QI*SCSF(ALPHA)-QALPH)*DADX(I)
C
C     BUOYANCY INCREMENTS.
C
C     BUOYANCY FORCE = - VELINF RHO A DV/DX    FOR FY, AND
```

```
C                               - VELINF RHO A DV/DX    FOR FZ.
C THUS, IN UNIFORM FLOW, BUOYANCY FORCE IS ZERO.
C
C
        R1=VELINF*RHO*ASTOR(I)
        DFYB=-R1*DVDX(I)
        DFZB=-R1*DWDX(I)
C
C       VISCOUS INCREMENTS.
C
        R1=0.5*RHO*ETACDC*SQRT(VI**2+WI**2)
        DFYV=(R1*VI-VYR)*D(I)
        DFZV=(R1*WI-VZR)*D(I)
        FYL(I)=DFYSB+DFYB+DFYV
5       FZL(I)=DFZSB+DFZB+DFZV
C       WRITE(9,123)FYL,FZL
C 123   FORMAT(/1X'FYL'/1X10G13.5/1X2G13.5//
C       1 1X'FZL'/1X10G13.5/1X2G13.5//)
C
C INTEGRATE W.R.T. X FROM X(NB) TO X(NF), TO GET TOTAL
C FORCE AND MOMENT INCREMENTS.
C
        DO8 I=1,2
        DO7 J=1,2
        FF=0.
        K=NB
6       CALL PARAB(ABC,X(K),FYZ(K,J))
        FF=FF+SUMF(X(K+2)-X(K))
        K=K+2
        IF(K.LE.NFM2) GO TO 6
        IF (K.EQ.NF) GO TO 7
        CALL PARAB(ABC,X(NFM2),FYZ(NFM2,J))
        FF=FF+SUMF(X(NF)-X(K))
7       DF(I,J)=FF
        DO8 L=1,NMS
        FNL(L)=X(L)*FYL(L)
8       FML(L)=-X(L)*FZL(L)
        QAM=QDPB*AMAX
        QAMD=QAM*DMAX
        DCY=DF(1,1)/QAM
        DCZ=DF(1,2)/QAM
        DCM=DF(2,2)/QAMD
        DCN=DF(2,1)/QAMD
        RETURN
        END
```

APPENDIX VIII

LISTINGS OF INTM AND DAUX

```
C INTM                                  DATE: 26-7-76
C
        SUBROUTINE INTM
C
C       CALLED BY MAIN.
C
C*     INTEGRATES 18 SIMULTANEOUS FIRST ORDER DIFFERENTIAL EQUATIONS.
C*     REQUIRES DIMENSION OF T TO BE 4*N+3 (=75)
C***
C*****N=NO OF EQUATIONS TO BE INTEGRATED
C
        COMMON T(320),IT(5)
        EQUIVALENCE (T(2),TIME),(T(3),DT)
C
C    ENTRY POINT FOR INTEGRATING ONE TIME STEP.
C*     CALLED BY MAIN.
C    FOURTH ORDER RUNGE KUTTA METHOD
C
        T(1)=TIME
        DO1 J=4,21
1       T(J+36)=T(J)
        DO 6 J=3,6
        A=(8-J)/2
        B=J/2
12      DO 66 I=4,21
        KN=I+36
        KO=I+54
        KP=I+18
        IF (J.EQ.6) GO TO 2
        T(KO)=T(KO)+DT*T(KP)*B
        T(I)=T(KN)+DT*T(KP)/A
        GO TO 66
2       T(I)=T(KN)+(DT*T(KP)+T(KO))/6.
        T(KO)=0.
66      CONTINUE
        TIME=T(1)+DT/A
        CALL DAUX
6       CONTINUE
        RETURN
        END
```

```
C DAUX                              DATE: 7-7-77
C
        SUBROUTINE DAUX
C
C       CALLED BY INTM.
C
C*    ROUTINE SPECIFIES THE EQUATIONS OF MOTION IN FIRST ORDER FORM SO
C*    THEY CAN BE INTEGRATED ONE TIME STEP BY INTM.
C*    FORCE EQUATIONS IN AIRCRAFT AXES.
C*    MOMENT EQUATIONS IN BOMB AXES.
C
        DIMENSION C(3,3),X(3),U(3),H(3),ABC(3,6),TIMES(3)
        DIMENSION DC(3,3),DX(3),DU(3),DH(3)
        DIMENSION FX(3),FL(3),A(3),P(3)
        DIMENSION AROT(3,3),BROT(3,3),ACC(3)
        COMMON T(320),IT(5)
        EQUIVALENCE (T(2),TIME)
        EQUIVALENCE (T(4),C),(T(13),X),(T(16),U),(T(19),H)
        EQUIVALENCE (T(22),DC),(T(31),DX),(T(34),DU),(T(37),DH)
        EQUIVALENCE (T(79),A),(T(82),STMASS)
        EQUIVALENCE (T(111),PA),(T(112),QA),(T(113),RA)
        EQUIVALENCE (T(114),FREE)
        EQUIVALENCE (T(117),ACC)
        EQUIVALENCE (T(120),FX),(T(123),FL)
        EQUIVALENCE (T(129),P),(T(130),Q),(T(131),R)
        EQUIVALENCE (T(137),AROT),(T(146),BROT)
        EQUIVALENCE (T(300),TIMES),(T(303),ABC)
C
        DELT=TIME-TIMES(1)
C*    RENORMALISE DIRECTION COSINES
        DO1 I=1,3
        SUM=0
C*****UPDATE ANGULAR VELOCITIES FROM ANGULAR MOMENTUM AT SAME TIME
        P(I)=H(I)/A(I)
        DO2 J=1,3
2       SUM=SUM+C(I,J)**2
        SUM=SQRT(SUM)
        DO1 J=1,3
1       C(I,J)=C(I,J)/SUM
C*    UPDATE BOMB AND AIRCRAFT ROTATION MATRICES
C
        IF (FREE.GT.0.) GO TO 7
        AROT(1,2)=RA
        AROT(1,3)=-QA
        AROT(2,1)=-RA
        AROT(2,3)=PA
        AROT(3,1)=QA
        AROT(3,2)=-PA
7       BROT(1,2)=R
        BROT(1,3)=-Q
        BROT(2,1)=-R
        BROT(2,3)=P(1)
        BROT(3,1)=Q
        BROT(3,2)=-P(1)
C*    EQUATION OF BOMB ORIENTATION W.R.T. AIRCRAFT.
C*    DC=BROT*C-C*AROT
C
```

```
C*     EQUATION OF BOMB C.G. MOTION REFERRED TO AIRCRAFT AXES.
C        XA=CA(XE-XE*)
C        DXA=CA(DXE-DXE*)+DCA(XE-XE*)
C
C        BUT DCA=AROT.CA
C
C        THEREFORE
C        DXA=CA(DXE-DXE*)+AROT.CA(XE-XE*)
C
C        LET U=CA(DXE-DXE*)
C
C        THEN
C        DXA=U+AROT.XA
C        AND
C            DU=CA(D2XE-D2XE*)+AROT.CA(DXE-DXE*)
C
C        I.E.
C                DU=CA.D2XE-CA.D2XE*+AROT.U
C
C        WHERE
C                CA.D2XE=FORCES/MASS
C                CA.D2XE*=AIRCRAFT ACCELERATIONS
C                        =(GFG.SIN(ATTACK),0,-GFG.COS(ATTACK))
C*     DX=U+AROT*X
C*     FORCE EQUATIONS OF MOTION,REFERRED TO AIRCRAFT AXES.
C*     DU=FX/STMASS-ACC+AROT*U
C*     MOMENT EQUATIONS OF MOTION,REFERRED TO BOMB AXES.
C*     DH=FL+BROT*H
C
         DO51 I=1,3
         DO51 J=1,3
         SUM=0.
         DO5 K=1,3
         SUM=SUM+BROT(I,K)*C(K,J)
         IF (FREE.LT.0.) SUM=SUM-C(I,K)*AROT(K,J)
5        CONTINUE
51       DC(I,J)=SUM
         DO6 I=1,3
         DX(I)=U(I)
         DU(I)=((ABC(3,I)*DELT+ABC(2,I))*DELT+ABC(1,I))/STMASS
         IF (FREE.LT.0.) DU(I)=DU(I)-ACC(I)
         I3=I+3
         DH(I)=(ABC(3,I3)*DELT+ABC(2,I3))*DELT+ABC(1,I3)
         DO6 J=1,3
         IF (FREE.GT.0.) GO TO 6
         DX(I)=DX(I)+AROT(I,J)*X(J)
         DU(I)=DU(I)+AROT(I,J)*U(J)
6        DH(I)=DH(I)+BROT(I,J)*H(J)
C        WRITE(9,44) DX,DU,DH
C 44     FORMAT(/1X'DX,DU,DH (DAUX)   ',9G12.5//)
C        WRITE(9,46) STMASS,DELT,H
C 46     FORMAT(1X'STMASS,DELT,H   ',5G12.5/)
C        WRITE(9,47) ((ABC(I,J),I=1,3),J=1,6)
C 47     FORMAT(1X'ABC ',3G15.6/(5X3G15.6))
         RETURN
         END
```

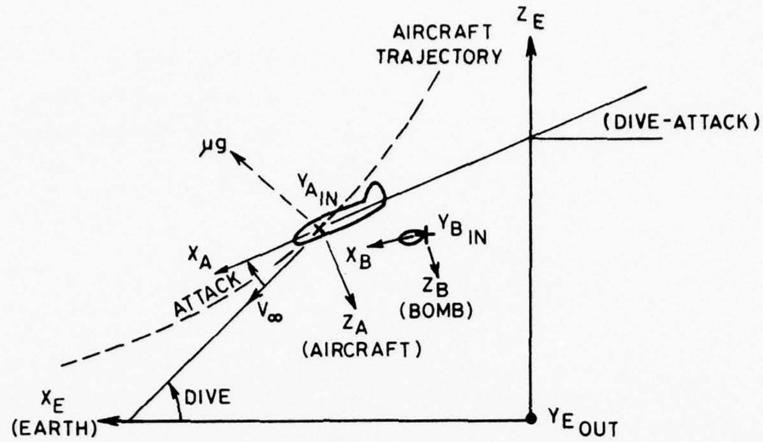

Figure 1. Definition of real world coordinate systems

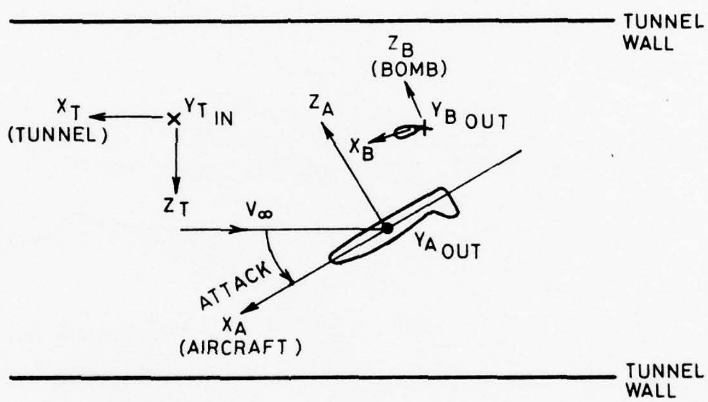

Figure 2. Definition of wind tunnel coordinate systems

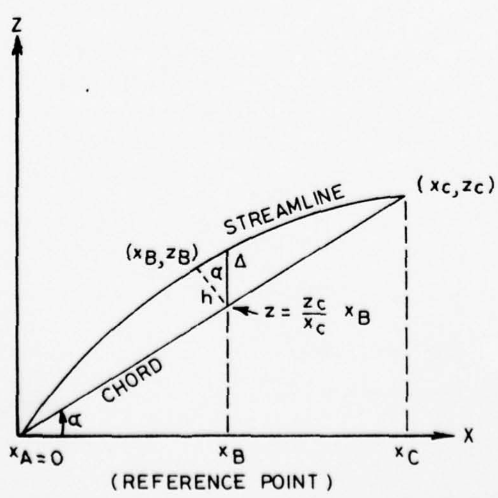

Figure 3. The camber of a curved stream

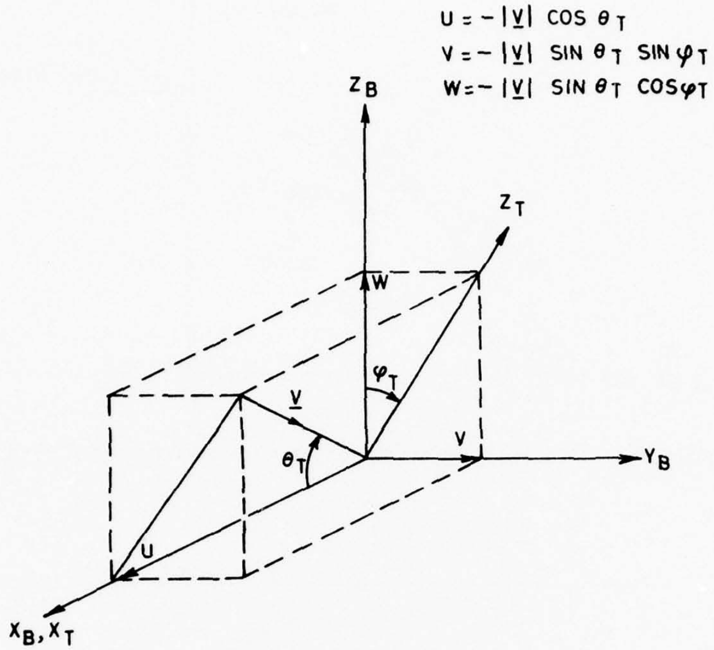

Figure 4. Definition of Wind Vector (V) Relative to Total (T) and Bomb (B) axes

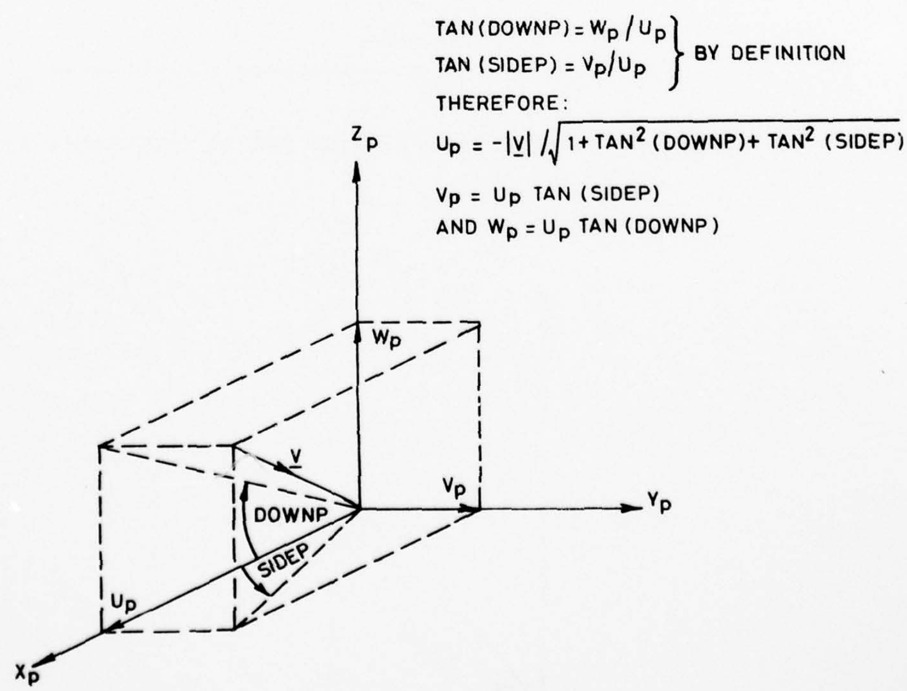

Figure 5. Definition of Wind Vector (V) Relative to Probe axes

DISTRIBUTION

EXTERNAL	Copy No.
In United Kingdom	
Defence Scientific and Technical Representative, London	1
Royal Aeronautical Establishment	
Aero Department: Farnborough	2
Bedford	3
Attention: Mr D.H. Perry	
Mr G. Brebner	
Mr P.G. Pugh	
Weapons Department: Library	4
Attention: Mr A. Smart	
Mr E.G. Cane	
R.A.R.D.E.: Library	5
National Lending Library of Science and Technology	6
T.T.C.P., U.K. National Leader, Panel W-2	7 - 10
Aeronautical Research Council	11
National Physical Laboratory, Teddington, Middlesex	12
Royal Aeronautical Society, Library	13
C.A.A.R.C., Secretary	14
Cambridge University, Engineering Department	
Attention: Professor W.A. Mair	15
Cranfield Institute of Technology	16
London University, Imperial College, Department of Aero. Engineering, Library	17
Oxford University, Department of Engineering Science	
Attention: Dr D.A. Spence	18
University of Bristol, Library	19
University of Southampton, Department of Aeronautics and Astronautics	
Attention: Professor G. Lilley	20
Aircraft Research Association (Bedford)	
Attention: Mr E.C. Carter	21
Society of British Aircraft Constructors	22
In United States of America	
Counsellor, Defence Science, Washington	23
Defence Research and Development Attache, Washington	24
T.T.C.P., US National Leader, Panel W-2	25
Air Force Armament Testing Laboratory	
Attention: Mr C.S. Epstein	26

	Copy No.
Applied Physics Laboratory, Johns Hopkins University	
Attention: Dr L. Cronvich	27
Arnold Engineering Development Center	
Attention: Dr M.L. Laster	28
Attention: Mr J.C. Marshall	29
Ballistics Research Laboratory	
Dr C.H. Murphy	30
Edgewood Arsenal, Library	31
Grumman Aerospace Corporation	32
N.A.S.A.	
Ames Research Center	
Attention: Dr G. Chapman	33
Langley Research Center, Library	
Attention: Mr J.E. Stitt	34
Lewis Research Center	35
Marshall Space Flight Center	
Attention: Mr J. Heaman	36
Picatinny Arsenal, Library	
Attention: Mr A. Loeb	37
Redstone Arsenal	
Attention: Mr R. Deep	38
Naval Surface Weapons Center	
White Oak: Mr S.M. Hastings	39
Dahlgren, Library	40
Naval Ship Research and Development Center, Library	41
Naval Weapons Center, Library	
Attention: Dr A.R. Maddox	42
Wright-Patterson Air Force Base, Library	43
American Institute of Aeronautics and Astronautics, Library	44
Applied Mechanics Reviews	45
Illinois Institute of Technology, Department of Mechanical and Aerospace Engineering, Library	46
University of California, Department of Mechanical Engineering, Library	47
A.R.O., Inc.	48
Sandia Corporation, Library	49

AD-A062 563 WEAPONS SYSTEMS RESEARCH LAB ADELAIDE (AUSTRALIA) F/G 19/4
THE SOFTWARE FOR THE CAPTIVE TRAJECTORY YAWMETER SYSTEM.(U)
APR 78 I C HERON, G R BISHOP
UNCLASSIFIED WSRL-0005-TR NL

2 OF 2
AD
A062563

END
DATE
FILMED
3-79
DDC

	Copy No.
In Canada	
D.R.E.V., Valcartier	50
N.A.E., Ottawa	51
T.T.C.P., Canadian National Leader, Panel W-2	52 - 55
In Europe	
A.G.A.R.D., Brussels	56
D.V.L., Germany	57
F.F.A., Sweden	58
N.L.L., Holland	59
O.N.E.R.A., France	60
Von Karman Institute for Fluid Dynamics, Belgium	61
In India	
Aeronautical Development Establishment, Bangalore	62
Indian Institute of Science, Bangalore (Department of Aero. Engineering)	
Attention: Professor R. Narasimha	63
Hindustan Aircraft Ltd., Bangalore	64
National Aeronautical Laboratories, Bangalore	
Attention: Dr A.V. Murthy	65
In Australia	
Chief Defence Scientist	66
Air Member for Technical Services	67
Director, Joint Intelligence Organisation (DDSTI)	68
Defence Central Library	69 - 70
Controller, Military Studies and Operational Analysis	71
Superintendent, Military Advisers	72
Navy Scientific Adviser	73
Army Scientific Adviser	74
Air Force Scientific Adviser	75
Superintendent, Central Studies Establishment	76
Controller, Program Policy and Planning	77
Head, International Programmes	78
Head, Defence Science and Technology Planning	79
Executive Controller, Australian Defence Scientific Services	80
Superintendent, Defence Science Administration	81

		Copy No.
Defence Information Services Branch for:		
United Kingdom, Ministry of Defence, Defence Research Information Centre (DRIC)		82
United States, Department of Defense, Defence Documentation Center		83 - 94
Canada, Department of National Defence, Defence Science Information Service		95
New Zealand, Department of Defence		96
Australian National Library		97
Controller, Service Laboratories and Trials		98
Superintendent, RAN Research Laboratory		99
Head, Engineering Development Establishment		100
Defence Library, Campbell Park		101
Aeronautical Research Laboratories		
Chief Superintendent		102
Superintendent, Aerodynamics Division		103
Superintendent, Mechanical Engineering Division		104
Library		105
Materials Research Laboratories		
Chief Superintendent		106
Library		107
Director of Research Requirements - Air Force		108
Aircraft Research and Development Unit, Edinburgh		
Head		109
OIC, Scientific Flight		110
Library		111
R.A.A.F. Academy, Pt Cook		
Warden		112
B.D.R.S.S., Canberra		113
Commonwealth Aircraft Corporation (Manager of Engineering)		114
Library, Government Aircraft Factory		115
Universities		
Adelaide	Barr Smith Library	116
	Professor of Mechanical Engineering	117
Australian National	Library	118
Melbourne	Library, Engineering School	119
Monash	Library	120
Newcastle	Library	121
New South Wales	Library, School of Mech. Engineering	122
Queensland	Library	123

		Copy No.
Universities		
Sydney	Library	124
	Professor of Aeronautical Engineering	125
Tasmania	Library, Engineering Department	126

INTERNAL

Chief Superintendent, Weapons Systems Research Laboratory	127
Superintendent, Aeroballistics Division	128
Head, Ballistics Composite	129
Principal Officer, Aerodynamic Research Group	130
Principal Officer, Dynamics Group	131
Principal Officer, Flight Research Group	132
Principal Officer, Ballistic Studies Group	133
Principal Officer, Field Experiments Group	134
Mr E.R.A. Landers, Aerodynamic Research Group	135
Dr G.H.S. Pike, Aerodynamic Research Group	136
Mr G.J. Simpson, Aerodynamic Research Group	137
Author(s)	138 - 139
AD Library	140 - 141
DRCS Library	142 - 143
Spares	144 - 163

US